Using semidirect products of groups to build classes of interconnection networks

Iain A. Stewart

Department of Computer Science, Durham University, Science Labs, South Road, Durham DH1 3LE, U.K.

Abstract

We build a framework within which we can define a wide range of Cayley graphs of semidirect products of abelian groups, suitable for use as interconnection networks and which we call toroidal semidirect product graphs. Our framework encompasses various existing interconnection networks such as cube-connected cycles, recursive cubes of rings, cube-connected circulants and dual-cubes, as well as certain multiswapped networks, pruned tori and biswapped networks; it also enables the construction of new hitherto uninvestigated but highly structured interconnection networks. We go on to design an efficient shortest-path routing algorithm that can be applied to any graph that can be defined within our framework. Our algorithm runs in time that is polylogarithmic in the size of the base group and polynomial in the size of the extending group of the given semidirect product. We also obtain analytic upper bounds on the diameters of our toroidal semidirect product graphs.

Keywords: interconnection networks, Cayley graphs, abelian groups, semidirect products, shortest path routing

1. Introduction

We begin with an explanation, for the uninitiated, why discrete mathematics has a role to play in the design of interconnection networks for parallel and distributed computing before highlighting the contributions of this paper.

1.1. Background

In a modern parallel or distributed computing system, such as a system-on-a-chip, a supercomputer or a data centre, the various individual components, such as processors, switches or servers, are interconnected via a point-to-point communications network. In order to support efficient communication, this network needs to possess a wide range of properties relating to, e.g., message routing,

Email address: i.a.stewart@durham.ac.uk (Iain A. Stewart)

message latency, message broadcasting, data throughput, scalability, fault tolerance, load balancing and ease of implementation. Some of these properties can work against each other which makes the design of these communications networks challenging. Moreover, because of their scale and cost, parallel or distributed computing systems cannot simply be built and tested; the efficiency of their communications networks needs to be validated prior to construction.

The abstraction of such a communications network as a graph is known as an *interconnection network*. More precisely, interconnection networks come in families in order to support scalability; that is, the facility to move to a larger network of the same type but so as to retain the same fundamental (point-to-point) routing algorithm. The standard example of a family of interconnection networks is the hypercubes (and their associated dimensional routing algorithm; see, e.g., [11]). Henceforth, by an interconnection network we mean a family of graphs (or possibly just an individual family member) specifically designed to support communication in parallel or distributed computing systems.

Regarding validation, it has long been recognised that certain mathematical properties of interconnection networks serve as good proxies for eventual performance. For example: the diameter of an interconnection network serves as a proxy for worst-case message latency; the bisection width can be used to estimate data throughput (under certain traffic patterns); high connectivity provides a framework for building fault-tolerant routing algorithms; multiple and judicious tree embeddings support message broadcasting and load balancing; and bounded degree assists with ease and cost of implementation. In addition, symmetry in an interconnection network, such as vertex-transitivity or edge-transitivity, is extremely beneficial when it comes to underpinning a range of performance-related properties. For instance, if we build a parallel or distributed system whose corresponding interconnection network is vertex-transitive then we can employ the same routing algorithm at any processor within the network and a rooted sub-network can be embedded so that any vertex can be chosen as the root. Examples of specific technical results relating to symmetry are: if an interconnection network is vertex-transitive (resp. regular and edge-transitive) then its edge-connectivity (resp. connectivity) is equal to its degree (and so is maximal; see [25, 26, 27, 35, 36]).

Whilst the study of interconnection networks is strongly motivated by their applications as communications networks, this study has its roots and a strong continued presence within discrete mathematics where structured graphs are investigated primarily as combinatorial objects (indeed, the study of many modern-day interconnection networks simply as graphs within discrete mathematics precedes the advent of computer communication). By 'structured' we mean graphs that can be described in a succinct way rather than via an adjacency matrix, say. Again, the hypercubes provide a good illustration: an n-dimensional hypercube has 2^n vertices yet there is an $O(n)$-time algorithm that given two vertices, that is, bit-strings of length n, outputs whether there is an edge joining these vertices. Such structured graphs are often built using combinatorial constructions such as graph products, typical of which are the cartesian, strong, direct, lexicographic, replacement and zig-zag products (a

glimpse at mathematical research on product graphs can be found in, e.g., [17]). Moreover, some of the combinatorial properties of interest with regard to structured graphs are inspired by, if not necessarily immediately directly relevant to, those of interconnection networks (see, e.g., [16, 17, 22]).

Suffice it to say, a plethora of interconnection networks has been proposed over the past fifty years or so (see, e.g., [19, 40]), the majority of which are defined using (product) constructions of discrete mathematics. As might be expected from our discussion above, *Cayley graphs* have featured heavily in interconnection network design, and continue to do so, for the algebra provides a means for succinct description and any Cayley graph is vertex-transitive (though not necessarily edge-transitive) which, as we hinted earlier, has many benefits in an interconnection network context. We refer the reader to the review papers [18, 20] for more on Cayley graphs and their relevance as interconnection networks and also note some more recent examples of research involving Cayley graphs in relation to interconnection network design: in a consideration of the structural properties of data centre interconnection networks with reference to universal routing schemes, it was shown in [10] that every Cayley graph has large hyperbolicity (hyperbolicity is a parameter that, intuitively, compares the metric space of a graph with the metric space of a tree); in [1], the computation of various paths in Cayley graphs was undertaken using automata theory (in the style of automatic groups [15]); and in [41], the generalized 3-connectivity of various classes of Cayley graphs was considered (generalized k-connectivity is a refined graph connectivity measure). These examples have been chosen as they involve path computation in Cayley graphs, as does our research; however, the general area encompassed by Cayley graphs and interconnection networks is thriving. In summary, not only are we interested in graph products but we are also interested in groups as mechanisms by which to design interconnection networks.

To sum up: we wish to build families of structured graphs that possess symmetry as well as myriad other properties supporting their use as communications networks; we want to use and develop tools and techniques from discrete mathematics to assist us; and we want to mathematically investigate a variety of properties of structured graphs purely as combinatorial objects.

1.2. Our contributions

In this paper, we are inspired by the extremely interesting construction by Mokhtar in [28] of the *cube-connected circulants* $CQ_n(d, r, m)$, where $n \geq 2$, $d, r \geq 3$ and m are positive integers such that $n \geq d$ and $dr \equiv 0 \pmod{n}$ (we define these graphs precisely later, as we do for all graphs mentioned in this introduction). The cube-connected circulants were in turn inspired by cube-connected cycles (due to Preparata and Vuillemin [30]) and recursive cubes of rings (due to Sun, Cheung and Lin [34] and subsequently refined by Mokhtar and Zhou [29]). The nature of these graphs is that they are 'hybrid', built as a 'product' of two other graphs; of course, the intention is to benefit by melding the attractive properties of both of the intrinsic graphs. In brief, cube-connected cycles are 'products' of cycles and hypercubes, as are recursive cubes of rings,

and cube-connected circulants are 'products' of multiplicative circulants and hypercubes. In order to make precise these vague notions of 'product', all these graphs can be defined as Cayley graphs of groups formed as *semidirect products* of direct products of cyclic groups by (that is, acted on by) cyclic groups.

Our first contribution is to rigorously generalize the construction of cube-connected cycles so that the class of 'constituent graphs' (corresponding to the multiplicative circulants in $CQ_n(d, r, m)$) is much broader as is the class of 'shape graphs' (corresponding to the hypercubes in $CQ_n(d, r, m)$). However, our graphs continue to be described as Cayley graphs of a semidirect product of two (abelian) groups. We call our general class of graphs *toroidal semidirect product graphs*. We explain how our framework not only encompasses cube-connected cycles, recursive cubes of rings and cube-connected circulants but also the dual-cubes from [23], (some of the) multiswapped networks from [32], pruned tori similar to those in [38], and (some of the) biswapped networks from [39]. Our framework is such that there is a range of parameters at our disposal so that we might vary the graphs obtained; indeed, there is also considerable scope for defining brand new classes of graphs.

Our second contribution is to design an efficient shortest-path routing algorithm that works no matter which groups and parameters are used to define a toroidal semidirect product graph. This algorithm runs in time that is polylogarithmic in the size of the base group and polynomial in the size of the extending group of the semidirect product defining our graph. Of course, our algorithm can be applied so as to yield an optimal routing algorithm for all of the classes of graphs mentioned in the previous paragraph.

In the next section, we present the basic definitions and concepts, whereas in Section 3 we describe our framework and explain why various classes of graphs from the literature sit within this framework. In Section 4, we briefly look at the connectivity of our toroidal semidirect product graphs before we provide the main proofs and technical constructions relating to our shortest-path routing algorithm in Sections 5 and 6: in Section 5, we reduce the search for shortest paths in our graphs to a search for specific 'covering' walks in a simplified graph, before examining these walks in detail in Section 6 and deriving our shortest-path routing algorithm. Our conclusions and directions for further research are given in Section 7.

2. Basic definitions, concepts and notation

The reader is referred to [9, 11, 12, 19, 40] for basic definitions and concepts from graph theory, interconnection networks, algorithms and group theory. We only include in this section material that is core to what follows.

We denote the identity of any group by 1_e and the order of any group element γ by $|\gamma|$ (it is always clear as to which group any identity element 1_e belongs). The cyclic subgroup generated by some element γ of some group $\mathcal{G}$ is denoted $\langle\gamma\rangle$. Given any group $\mathcal{G}$ and set of distinct elements $\Gamma_{\mathcal{G}} \subseteq \mathcal{G} \setminus \{1_e\}$, where $\Gamma_{\mathcal{G}}$ is closed under inverses, we define the *Cayley graph* $Cay(\mathcal{G}; \Gamma_{\mathcal{G}})$ to have the set of elements of $\mathcal{G}$ as its set of vertices and an edge joining some vertex $g \in \mathcal{G}$ to

the vertex $g\gamma$, for every $\gamma \in \Gamma_{\mathcal{G}}$. Note that any Cayley graph is simple, as are all graphs in this paper.

Let $\mathcal{G}$ be a group and let Ω be a set. Suppose that for every $g \in \mathcal{G}$, there is a permutation φ_g of Ω so that: φ_{1_e} is the identity permutation; and $\varphi_{gh} = \varphi_g\varphi_h$. Then we say that $\mathcal{G}$ *acts* on Ω or that the permutations $\varphi = \{\varphi_g : g \in \mathcal{G}\}$ form an *action* of $\mathcal{G}$ on Ω. Suppose that Ω consists of the elements of a group $\mathcal{Q}$. We say that $\mathcal{G}$ *acts* on $\mathcal{Q}$ if $\mathcal{G}$ acts on $\mathcal{Q}$ as a set and also the action respects the group structure of $\mathcal{Q}$; that is, for each $g \in \mathcal{G}$ and $q_1, q_2 \in \mathcal{Q}$, we have that $\varphi_g(q_1q_2) = \varphi_g(q_1)\varphi_g(q_2)$ (that is, φ_g is an automorphism of $\mathcal{Q}$). Let $\mathcal{G}$ and $\mathcal{Q}$ be groups so that $\mathcal{G}$ acts on $\mathcal{Q}$ via the action φ. The *semidirect product* $\mathcal{Q} \rtimes \mathcal{G}$ is the group whose element set is $\mathcal{Q} \times \mathcal{G}$ and where the group multiplication is defined via $(q, g)(q', g') = (q\varphi_g(q'), gg')$, for all $g, g' \in \mathcal{G}$ and $q, q' \in \mathcal{Q}$. The group $\mathcal{Q}$ is referred to as the *base group* and the group $\mathcal{G}$ as the *extending group* of the semidirect product $\mathcal{Q} \rtimes \mathcal{G}$.

A *circulant* on $n \geq 2$ vertices is a graph that is a Cayley graph of the cyclic group $\mathbb{Z}_n$ of order n. So, a circulant can be thought of as having: the vertex set $\{0, 1, \ldots, n-1\}$; an associated set $I \subseteq \{1, 2, \ldots, n-1\}$ so that $j \in I$ if, and only if, $n - j \in I$; and where for any distinct $u, v \in \{0, 1, \ldots, n-1\}$, (u, v) is an edge if, and only if, $|u - v| \in I$. A *multiplicative circulant* [33] is a circulant graph where $n = r^m$, for some $r \geq 2$ and $m \geq 1$, and where $I = \{r^i, r^m - r^i : i = 0, 1, \ldots, m-1\}$ (with any repetitions removed). An *n-dimensional torus*, where $n \geq 1$, is the Cartesian product of n cycles; if the length of each of these cycles is $k \geq 3$ then we obtain the *k-ary n-cube* Q_n^k (we can regard hypercubes as special 2-ary n-cubes). For any graph G and any two vertices u and v of G, we denote the length of a shortest path in G from u to v by $d_G(u, v)$ and we denote the diameter of G by $diam(G)$.

As we will hear later, low degree and low diameter feature in our toroidal semidirect product graphs and it is worthwhile providing a little more information as to why these qualities are required in interconnection networks. Concerning the degree, first, if we fix the number of vertices in a graph then the lower the degrees of the vertices, the fewer wires or cables we will need when we build the corresponding interconnection network. Second, each of the routers associated with a processor has a limited number of pins with these pins partitioned so as to be allocated to channels. Securing a higher bandwidth for the resulting channels means having a smaller number of channels; that is, the smaller the degree, the higher the potential channel bandwidth. Third, the more incoming and outgoing channels to some router (within some processor), the more complex the router architecture needs to be so as to be able to handle the additional traffic and, also, the higher the latency of the router (that is, the time to get data across the router). We have already heard that the diameter serves as a proxy for worst-case message latency. The combination of low degree and low diameter motivates the study of *Moore graphs* as potential interconnection networks (see, e.g., [7]).

3. Building our interconnection networks

We now describe a general construction of an interconnection network formed as a Cayley graph of a semidirect product $\mathcal{S} = \mathcal{Q} \rtimes \mathcal{G}$. As we shall see, there are various conditions on and parameters associated with the groups $\mathcal{Q}$ and $\mathcal{G}$ that make them viable with respect to our construction. We illustrate our methodology by incrementally building the cube-connected circulants as we proceed; as we shall see, this involves choosing $\mathcal{Q}$ to be the direct product of cyclic groups of order 2 and $\mathcal{G}$ to be a multiplicative circulant (together with a specific action of $\mathcal{G}$ on $\mathcal{Q}$, a specific generating set of $\mathcal{Q} \rtimes \mathcal{G}$ and other associated parameters).

3.1. The group $\mathcal{G}$

Let $\mathcal{G} = \mathcal{H} \times \langle \gamma \rangle$ be a finite abelian group where $|\gamma| = cr$, with $c \geq 1$ and $r \geq 2$. Let

$$\Gamma_{\mathcal{G}} = \{(1_e, \gamma), (1_e, \gamma^{-1})\} \cup \{(1_e, \gamma^{kr}) : k \in I_R\} \cup \{(\sigma, 1_e) : \sigma \in \Gamma_{\mathcal{H}}\},$$

where:

- $I_R \subseteq \{1, 2, \ldots, c-1\}$ so that $k \in I_R$ if, and only if, $c - k \in I_R$
- if $cr = 2$ then $(1_e, \gamma^{-1})$ is omitted from $\Gamma_{\mathcal{G}}$
- $\Gamma_{\mathcal{H}}$ is an inverse-closed set of non-trivial elements of the group $\mathcal{H}$ that generates $\mathcal{H}$.

Consequently, $\Gamma_{\mathcal{G}}$ is inverse-closed and generates $\mathcal{G}$. Define the graph $G = Cay(\mathcal{G}; \Gamma_{\mathcal{G}})$. It will be useful to describe G as follows.

- The subgraph G_0 defined as:

  $$G_0 = Cay(1_e \times \langle \gamma \rangle; \{(1_e, \gamma), (1_e, \gamma^{-1})\} \cup \{(1_e, \gamma^{kr}) : k \in I_R\})$$

  (with the generator $(1_e, \gamma^{-1})$ omitted if $cr = 2$) is isomorphic via the map $(1_e, \gamma^i) \mapsto i \pmod{cr}$ to the graph, whose vertex set is $\{0, 1, \ldots, cr-1\}$, that is formed by a cycle $0, 1, 2, \ldots, cr-1, 0$, which we call the *basic cycle* of G_0, so that in addition each vertex i is adjacent to vertex $kr + i \pmod{cr}$, for each $k \in I_R$; consequently, G_0 is a circulant graph. (Note that if $cr = 2$ then G_0 is just an edge rather than a cycle.)

  - For each $0 \leq i \leq r-1$, we can think of the vertices of $\{jr + i : j = 0, 1, \ldots, c-1\}$ as lying on 'row' i of G_0 with an edge $(j_1 r + i, j_2 r + i)$ between two distinct vertices on row i if, and only if, either $j_1 - j_2$ or $c - (j_1 - j_2)$ lies in I_R. Hence, the vertices on any row i induce a graph isomorphic to the (circulant) graph R on the vertices $\{0, 1, \ldots, c-1\}$ via the natural isomorphism $jr + i \mapsto j$.

– There are edges from $\{(i, i+1) : 0 \leq i \leq cr-1\}$ (with addition modulo cr; these are edges of the basic cycle) lying in 'columns' so that there are 'wrap-around' edges from the bottom of column j to the top of column $j+1$, for $j = 0, 1, \ldots, c-1$, if $c > 1$ (addition is modulo c). If $c = 1$, we have that: if $r > 2$ then there is a wrap-around edge from the bottom of column 0 to the top of column 0; and if $r = 2$ then the two column vertices form an edge.

– Hence, for each $0 \leq i \leq cr-1$, we can rename the vertex $i = yr + x$, where $0 \leq x \leq r-1$ and $0 \leq y \leq c-1$, as (x, y) whence we have column-edges

$$\{((x,y),(x+1,y)) : 0 \leq x \leq r-2; 0 \leq y \leq c-1\}$$
$$\cup\{((r-1,y),(0,y+1 \pmod c)) : 0 \leq y \leq c-1\}$$

and row-edges

$$\{((x,y),(x,y+k \pmod c))) : 0 \leq x \leq r-1; 0 \leq y \leq c-1; k \in I_R\}.$$

- The graph G is formed from $|\mathcal{H}|$ copies of G_0, each labelled with a unique element of $\mathcal{H}$; so, the vertices of G can also be named $\{(\sigma, x, y) : \sigma \in \mathcal{H}; 0 \leq x \leq r-1; 0 \leq y \leq c-1\}$ in the obvious way. Throughout, we regard vertices of G as group elements and denote them as (σ, γ^i) or (σ, x, y) interchangeably.

- For any $0 \leq x \leq r-1$ and $0 \leq y \leq c-1$, there are some additional edges of the form $((\sigma, x, y), (\sigma', x, y))$, where $\sigma \neq \sigma'$. For any fixed such x and y, these additional edges induce a graph isomorphic to $Cay(\mathcal{H}; \Gamma_{\mathcal{H}})$ on the vertices of $\{(\sigma, x, y) : \sigma \in \mathcal{H}\}$. We call these additional edges $\mathcal{H}$-edges.

The graph G_0 of the copy of G labelled $\sigma \in \mathcal{H}$ can be visualized as in Fig. 1 (here, $c > 1$) and the graph G in Fig. 2. Note that the edge-set of G partitions as the set of row-edges, the set of column-edges and the set of $\mathcal{H}$-edges. The parameters at our disposal when defining $\mathcal{G}$ are: r and c; I_R; and $\mathcal{H}$ and $\Gamma_{\mathcal{H}}$.

In the definition of the cube-connected circulants in [28], when building $\mathcal{G}$ we choose: $r \geq 3$; $c = r^{p-1}$, for some $p \geq 1$; $I_R = \{r^i, r^{p-1} - r^i : i = 0, 1, \ldots, p-2\}$ (with any repetitions removed); and $\mathcal{H}$ to be the trivial group. Consequently, G is the multiplicative circulant on r^p vertices.

3.2. The group $\mathcal{Q}$ acted on by $\mathcal{G}$

Let $\mathcal{Q}$ be a finite direct product of finite cyclic groups. Batch these groups together so that batch B_k is the direct product of $n_k \geq 1$ cyclic groups $\mathbb{Z}_{b_k}$, where $b_k \geq 2$, for $1 \leq k \leq m$, and where $m \geq 1$ and $b_1 > b_2 > \ldots > b_m$. Suppose that for $1 \leq k \leq m$, the generators of the groups in batch B_k are $q_{k,1}, q_{k,2}, \ldots, q_{k,n_k}$. Define $n = \sum_{k=1}^{m} n_k$. We can consider an element

$$\mathbf{q} = (q_{1,1}^{a_{1,1}}, q_{1,2}^{a_{1,2}}, \ldots, q_{1,n_1}^{a_{1,n_1}}, q_{2,1}^{a_{2,1}}, q_{2,2}^{a_{2,2}}, \ldots, q_{2,n_2}^{a_{2,n_2}}, \ldots, q_{m,1}^{a_{m,1}}, q_{m,2}^{a_{m,2}}, \ldots, q_{m,n_m}^{a_{m,n_m}})$$

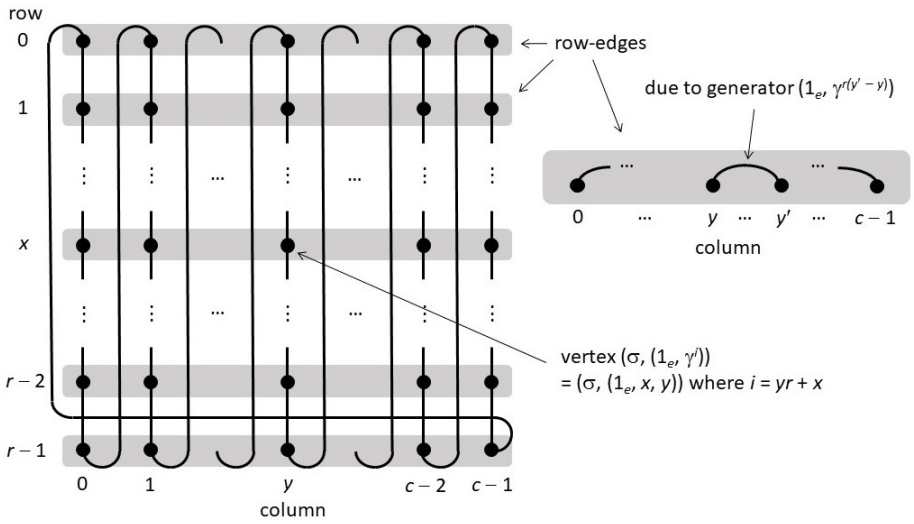

Figure 1: The graph G_0.

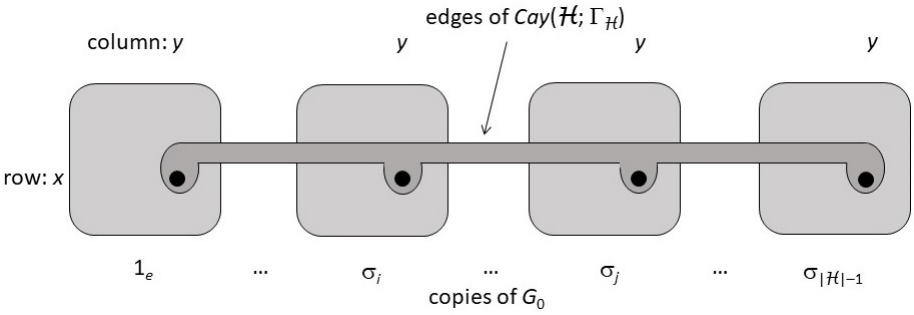

Figure 2: The graph G.

of $\mathcal{Q}$ as an n-tuple of integers

$$(a_{1,1}, a_{1,2}, \ldots, a_{1,n_1}, a_{2,1}, a_{2,2}, \ldots, a_{2,n_2}, \ldots, a_{m,1}, a_{m,2}, \ldots, a_{m,n_m}).$$

Fix $1 \leq k \leq m$. Let M_k be an $n_k \times n_k$ permutation matrix (that is, there is exactly one 1 in every row and column, with 0s elsewhere) so that the order $|M_k|$ of M_k divides r, where r is the parameter from our construction of $\mathcal{G}$ in Section 3.1. The group $\langle M_k \rangle$ acts, by left multiplication, on the set of column vectors $\{\mathbf{e}_i : i = 1, 2, \ldots, n_k\}$, where $\mathbf{e}_i \in \{0,1\}^{n_k}$ has 1 in the ith component and 0 elsewhere. When choosing M_k, also choose $1 \leq d_k \leq n_k$ and ensure that the orbit of $\{\mathbf{e}_i : i = 1, 2, \ldots, d_k\}$ under $\langle M_k \rangle$ is $\{\mathbf{e}_i : i = 1, 2, \ldots, n_k\}$; so, $n_k \leq d_k |M_k|$. Define $|M_k| > \mu_k \geq 0$ to be the least such μ_k so that $\{\mathbf{e}_i : i = 1, 2, \ldots, n_k\} = \{M_k^l \mathbf{e}_j : 1 \leq j \leq d_k; 0 \leq l \leq \mu_k\}$; so, $n_k \leq d_k(\mu_k + 1)$. Define $d = \sum_{k=1}^{m} d_k$. The value of μ_k will obviously depend on the cycle decomposition

of the permutation p_k described by M_k and the chosen value of d_k. In any case, $|M_k|$ is the least common multiple of the lengths of the cycles in the cycle decomposition of p_k, and μ_k is less than the length of the longest cycle in the cycle decomposition of p_k (certainly, $\mu_k < n_k$ and $\mu_k < r$).

Define $\mu = \max\{\mu_k : 1 \leq k \leq m\}$ and M to be the $n \times n$ matrix obtained by placing $M_1, M_2, \ldots, M_m$ down the leading diagonal, with 0s elsewhere. Consequently, we have that: $|M| = \text{lcm}\{|M_k| : 1 \leq k \leq m\}$ divides r; $|M|$ is the least common multiple of the lengths of all cycles of the cycle decompositions of the permutations $p_1, p_2, \ldots, p_m$; and μ is less than the length of the longest cycle of the cycle decompositions of the permutations $p_1, p_2, \ldots, p_m$ (certainly, $n \leq d(\mu+1)$, $\mu < n$ and $\mu < r$). The matrix M can be visualized as in Fig. 3(a). There is an obvious action of $\langle M \rangle$ on $\mathcal{Q}$ by left multiplication.

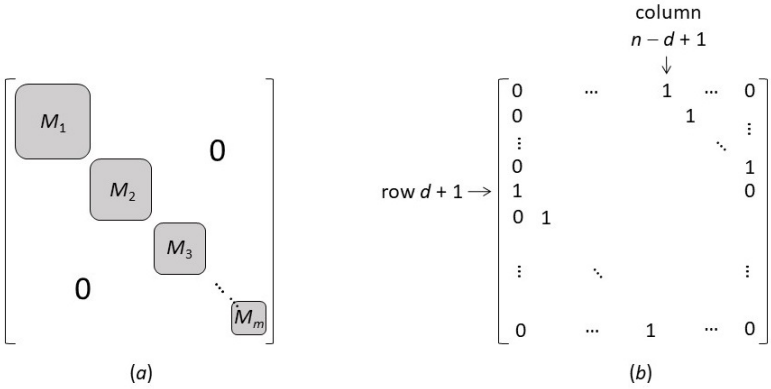

Figure 3: The $n \times n$ matrix M in general and for cube-connected circulants.

Consider the following action of $\mathcal{G}$ on $\mathcal{Q}$: for $g = (\sigma, \gamma^i) \in \mathcal{G}$, where $(\sigma, \gamma^i) = (\sigma, x, y) \in G$, and for $\mathbf{q} \in \mathcal{Q}$, define the map $\varphi_g : \mathbf{q} \mapsto M^i\mathbf{q}$. It is trivial to check that this is indeed an action of $\mathcal{G}$ on $\mathcal{Q}$. Note that because $|M|$ divides r, we have that for any integer i, (σ, γ^i) acts like $(\sigma, \gamma^{i \pmod r}) = (\sigma, \gamma^x)$ on $\mathcal{Q}$. Consequently, for any $0 \leq x \leq r-1$, all elements of G lying in row x of any copy of G_0 within G act on $\mathcal{Q}$ in the same way (note also that depending upon the values of all the parameters involved, it might be the case that elements lying on rows x and x', where $0 \leq x < x' \leq r-1$, of possibly different copies of G_0 within G act on $\mathcal{Q}$ in the same way). The parameters at our disposal when defining $\mathcal{H}$ are: m; $b_1, b_2, \ldots, b_m$; $n_1, n_2, \ldots, n_m$; $M_1, M_2, \ldots, M_k$; and $d_1, d_2, \ldots, d_m$, but under the stipulations that $|M_k|$ divides r and the orbit of $\{\mathbf{e}_i : i = 1, 2, \ldots, d_k\}$ under $\langle M_k \rangle$ is $\{\mathbf{e}_i : i = 1, 2, \ldots, n_k\}$, for $1 \leq k \leq m$.

When building the cube-connected circulants in [28], with regard to $\mathcal{Q}$ we choose: $m = 1$; $b_1 = 2$; $n_1 = n$; $d_1 = d$, such that n divides rd; and M to be the $n \times n$ permutation matrix as depicted in Fig. 3(b) (corresponding to the permutation that cyclically shifts vector components d places to the right with 'wrap-around'). Hence, $\mu = \mu_1 = \lceil \frac{n}{d} \rceil - 1$, $\mathcal{Q}$ is the direct product of n copies of $\mathbb{Z}_2$ and because of our stipulation that n divides rd, $|M|$ divides r.

3.3. Our interconnection network

We are now in a position to build our interconnection network S. Form the semidirect product $\mathcal{S} = \mathcal{Q} \rtimes \mathcal{G}$ (using the above action of $\mathcal{G}$ on $\mathcal{Q}$). For any $1 \leq k \leq m$ and $1 \leq j \leq n_k$, define

$$\mathbf{q}_{k,j}^{\epsilon} = (0, \ldots, 0, \epsilon, 0, \ldots, 0) \in \mathcal{Q},$$

where $\epsilon = \pm 1$ lies in component $j + \sum_{i=1}^{k-1} n_i$ (with the elements of $\mathcal{Q}$ taken as n-tuples of integers), and define

$$\Gamma_{\mathcal{Q}} = \bigcup_{k=1}^{m} \{\mathbf{q}_{k,1}^{\epsilon}, \mathbf{q}_{k,2}^{\epsilon}, \ldots, \mathbf{q}_{k,d_k}^{\epsilon} : \epsilon = \pm 1\},$$

where if $b_m = 2$ then we remove all duplications of the form $\mathbf{q}_{m,j}^{-1}$ from $\Gamma_{\mathcal{Q}}$. Note that $\bigcup_{l=0}^{\mu} \{M^l \mathbf{q} : \mathbf{q} \in \Gamma_{\mathcal{Q}}\} = \{\mathbf{q}_{k,j}^{\epsilon} : 1 \leq k \leq m; 1 \leq j \leq n_k; \epsilon = \pm 1\}$.

Define $S = Cay(\mathcal{Q} \rtimes \mathcal{G}; (1_e \times \Gamma_{\mathcal{G}}) \cup (\Gamma_{\mathcal{Q}} \times 1_e))$. Hence, S consists of $|\mathcal{Q}|$ disjoint copies of G with additional edges dictated by $\Gamma_{\mathcal{Q}}$ and the action of $\mathcal{G}$ on $\mathcal{Q}$; we call these additional edges the $\mathcal{Q}$-edges of S.

The cube-connected circulant $CQ_n(d, r, m)$ is $Cay(\mathcal{Q} \rtimes \mathcal{G}; (1_e \times \Gamma_{\mathcal{G}}) \cup (\Gamma_{\mathcal{Q}} \times 1_e))$ where $\mathcal{G}$ is as in the final paragraph of Section 3.1 and $\mathcal{Q}$ is as in the final paragraph of Section 3.2. In general, we call any graph S constructed as in this section a *toroidal semidirect product graph*.

3.4. Our graph S as an interconnection network

Intuitively speaking, the graph S consists of copies of G, one for each element of $\mathcal{Q}$. A vertex in some copy of G is adjacent to d vertices of the same name in other copies of G with the actual copies of G dictated by the action of $\mathcal{G}$ on $\mathcal{Q}$ and our choice of parameters. Some basic properties of S are as follows:

- the number of vertices in S is $|\mathcal{G}||\mathcal{Q}| = |\mathcal{H}|cr \prod_{k=1}^{m} b_k^{n_k}$
- S is regular of degree $|\Gamma_{\mathcal{H}}| + |I_R| + d_{\mathcal{G}} + d_{\mathcal{Q}}$ where:
  - if $b_m > 2$ then $d_{\mathcal{Q}} = 2 \sum_{j=1}^{m} d_j$
  - if $b_m = 2$ then $d_{\mathcal{Q}} = d_m + 2 \sum_{j=1}^{m-1} d_j$
  - if $cr > 2$ then $d_{\mathcal{G}} = 2$
  - if $cr = 2$ then $d_{\mathcal{G}} = 1$.

One of the motivations for Mokhtar to introduce the cube-connected cycles in [28] was the need to improve the path-length deficiencies of cycles, in recursive cubes of rings, by using multiplicative circulants instead. In turn, one of the motivations for the study of the recursive cubes of rings in [34] was to enable scalability and the beneficial properties of hypercubes but so as to keep the degree constant and exert control over the diameter. Our toroidal semidirect product graphs provide for these controls but are much more general than what has gone before (as we'll see in the next section).

As we have explained above, our graph S, of size $|\mathcal{G}||\mathcal{Q}| = |\mathcal{G}|b^n$, for some $b_1 \leq b \leq b_m$, is intended to have beneficial properties in relation to its potential usage as an interconnection network. A necessary property of any interconnection network is that there is an efficient routing algorithm, where 'efficient' means of low time and space complexity and easy to implement (remember: a routing algorithm needs to be implemented at each processor of a distributed system). In particular, given the fact that interconnection networks are intended to have hundreds of thousands or even millions of processors, hosting a complete representation of the (adjacency matrix of the) interconnection network at each processor is not feasible as the memory required to host routing tables at each processor can be prohibitive. Also, routing tables are much more inflexible than a routing algorithm when it comes to tolerating faults or enabling adaptive route selection. With regard to S, we think of the graph G, of size $|\mathcal{G}|$, as being small enough to deal with explicity but we insist that any routing algorithm should have time complexity that is polynomial in n (the 'dimension' of our 'shape graph'). We say that a routing algorithm for a toroidal semidirect product graph has time complexity that is *polynomial* in $|\mathcal{G}|$ and *polylogarithmic* in $|\mathcal{Q}|$ if the time complexity is polynomial in both $|\mathcal{G}|$ and n.

3.5. Some examples

We have already explained how the cube-connected circulants are defined within our framework. Let us now describe some other classes of graphs that can be so defined (we end with some classes of graphs from the literature which are built around semidirect products but do not quite fit within our framework).

The *cube-connected cycles* CCC_n, for $n \geq 3$, originated in [30] and has vertex set $\{0,1\}^n \times \{0,1,\ldots,n-1\}$ where there is an edge from $(\mathbf{u}, i)$ to $(\mathbf{v}, j)$, for any $\mathbf{u}, \mathbf{v} \in \{0,1\}^n$ and for any $0 \leq i, j \leq n-1$, if one of the following two cases holds:

- $\mathbf{u} = \mathbf{v}$ and $|i - j| = 1 \pmod{n}$
- $i = j$ and $\mathbf{u}$ and $\mathbf{v}$ differ only in the $(i+1)$th bit.

Suppose that when building $\mathcal{G}$ we choose: $r \geq 3$ and $c = 1$; $I_R = \emptyset$; and $\mathcal{H}$ as the trivial group. Hence, G is a cycle of length $r \geq 3$. With regard to $\mathcal{Q}$, we choose: $m = 1$; $b_1 = 2$; $n_1 = r$; and $d_1 = 1$. Consequently, $\mathcal{Q}$ is the direct product of r copies of $\mathbb{Z}_2$. The $r \times r$ matrix M is the permutation matrix that cyclically shifts each component of a vector one place to the right, with 'wrap-around', and can be visualized as in Fig. 3(b) except that r replaces n in the figure with 1 replacing d. The resulting graph S is the cube-connected cycles CCC_r and is a special case of a cube-connected circulant.

Another special case of a cube-connected circulant is the *recursive cubes of rings*. Recursive cubes of rings were first defined in [34] but not as Cayley graphs and restrictions need to be applied, as was done in [29], so that they can be realised as Cayley graphs of semidirect products. They are defined within our framework just as cube-connected circulants are except that in addition

$c = 1$; so, G is a cycle rather than a multiplicative circulant, as is the case for cube-connected circulants.

Biswapped networks were proposed in [39] as 'symmetric versions' of *OTIS networks*, which originated as models of optoelectronic interconnection networks (see [31] for more on the history and evolution of such networks). The biswapped network $Bsw(G)$, where G is a graph on the vertex set $\{v_1, v_2, \ldots, v_n\}$, is defined by taking $2n$ copies of G, labelled by elements of $\{0,1\} \times \{v_1, v_2, \ldots, v_n\}$, and including the edge joining vertex v_i of the copy of G labelled $(0, v_j)$ to the vertex v_j of the copy of G labelled $(1, v_i)$, for all $i, j \in \{v_1, v_2, \ldots, v_n\}$. A visualization of $Bsw(G)$ is given in Fig. 4 where the copies of G labelled $(0, v_i)$ (where the index is v_i in the figure) are pictured at the top and those labelled $(1, v_i)$ at the bottom. Suppose we take G to be the cycle C_p of length $p \geq 3$. One can see in Fig. 4 the 'shape' of a three-dimensional torus. The biswapped network $Bsw(C_p)$ can be defined within our framework as follows. With regard to $\mathcal{G}$, choose: $r = 2$ and $c = 1$; $I_R = \emptyset$; and $\mathcal{H}$ to be the trivial group. So, G is an edge. With regard to $\mathcal{Q}$, choose: $m = 1$; $b_1 = p$; $n_1 = 2$; $d_1 = 1$; and the 2×2 matrix M to be $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. Consequently, $\mathcal{Q}$ is $\mathbb{Z}_p \times \mathbb{Z}_p$, with the resulting graph S isomorphic to $Bsw(C_p)$. Of course, other biswapped networks $Bsw(G)$ can be defined within our framework. For example, if we choose: $r = 2$ and $c = 1$; $I_r = \emptyset$; $\mathcal{H}$ to be the trivial group; $m = 1$; $b_1 = p$; $n_1 = 4$; $d_2 = 2$; and the 4×4 matrix M to be $\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$ then we obtain that the resulting graph S is isomorphic to the biswapped network $Bsw(Q_2^p)$.

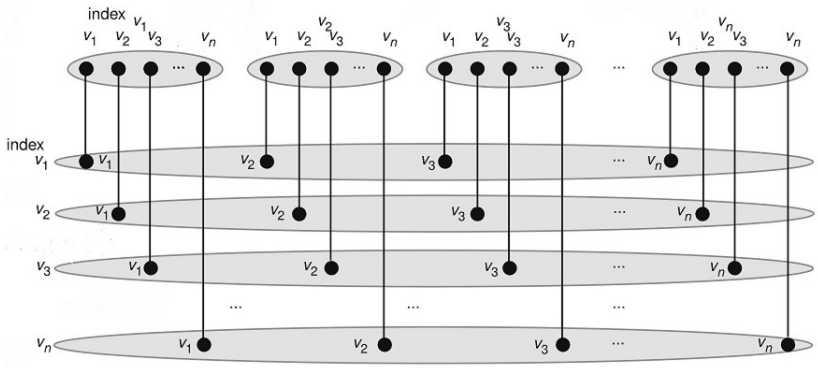

Figure 4: The graph $Bsw(G)$.

Suppose that when building $\mathcal{G}$ we choose: $r \geq 4$ is even and $c = 1$; $I_R = \emptyset$; and $\mathcal{H}$ as the trivial group. Hence, G is a cycle of even length $r \geq 4$. With regard to $\mathcal{Q}$, we choose: $m = 1$; $b_1 = r$; $n_1 = 2$; and $d_1 = 1$. Consequently, $\mathcal{Q}$ is

$\mathbb{Z}_r \times \mathbb{Z}_r$. The 2×2 matrix M is defined as $\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$. The resulting graph S can be regarded as a 'pruned' r-ary 3-cube Q_3^r:

- a vertex of the form $(\mathbf{q}, (\sigma, \gamma^i))$ can be identified as $((u, v), (1_e, x, 0))$, i.e., as (u, v, x), with u (resp. v, x) visualized three-dimensionally as indexing the place of a vertex in a left-to-right row (resp. front-to-back row, column)
- the r vertices in the column indexed as $(u, v, *)$ are joined in a cycle of length r
- if x is even (resp. odd) then the r vertices in the left-to-right (resp. front-to-back) row indexed as $(*, v, x)$ (resp. $(u, *, x)$) are joined in a cycle of length r.

The graph S is actually identical to the *multiswapped network* $Msn(C_r; C_r)$ of [31], where C_r is a cycle of length r (the graph $Msn(C_r; C_r)$ can be pictured similarly to that in [32, Fig. $4(a)$]). Multiswapped networks are essentially built by 'fusing' together lots of biswapped networks. It is easy to extend the above construction so that we can build a variety of pruned k-ary n-cubes within our framework.

The *dual-cube* DC_n, for $n \geq 1$, originates in [23] and is defined there as follows:

- the vertex set is $\{0, 1\}^{2n+1}$
- there is an edge joining two vertices if, and only if,
  - the bit-strings of the two vertices differ in exactly one position, with the additional proviso that
  - if the position where the bit-strings differ is position i and $1 \leq i \leq n$ (resp. $n + 1 \leq i \leq 2n$) then position $2n + 1$ of both bit-strings is necessarily 0 (resp. 1).

When building $\mathcal{G}$, we choose: $r = 2$ and $c = 1$; $I_R = \emptyset$; and $\mathcal{H}$ to be the trivial group. So, G is an edge. With regard to $\mathcal{Q}$, we choose: $m = 1$; $b_1 = 2$; $n_1 = 2p$; and $d_1 = p$. Consequently, $\mathcal{Q}$ is the direct product of $2p$ copies of $\mathbb{Z}_2$. The $2p \times 2p$ matrix M is the permutation matrix that swaps the ith and $(p + i)$th components of a vector, for every $1 \leq i \leq p$; this matrix can be obtained from the matrix above in the multiswapped network case by replacing 0 in that matrix with a $p \times p$ matrix of 0s and 1 with a $p \times p$ identity matrix. The graph S is the dual-cube DC_p. (Note that DC_p was observed to be a Cayley graph of the semidirect product $\mathcal{Q} \rtimes \mathcal{G}$ in [42]; however, the definition of DC_p as such on [42, p. 1734] is incorrect.)

Consider the following more complex construction. When building $\mathcal{G}$, we choose: $r \geq 3$ divisible by 3; $c \geq 6$; $I_R = \{2, 5\}$; and $\mathcal{H}$ to be the trivial group. The graph G is a circulant and can be visualized as in Fig. $5(a)$ when $r = 6$ and $c = 7$ (the column-edges are depicted as solid lines and the row-edges

are depicted as dashed lines). With regard to $\mathcal{Q}$, we choose: $m = 1$; $b_1 = 2$; $n_1 = 3$; and $d_1 = 2$. Consequently, $\mathcal{Q}$ is the direct product of 3 copies of $\mathbb{Z}_2$. We choose the 3×3 matrix M as $\begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$. So, the resulting graph S consists of 8 copies of G where: all vertices in row 0 in any copy of G labelled $(\alpha_1, \alpha_2, \alpha_3) \in \{0,1\}^3$ are adjacent to the vertices of the same name in the copies of G labelled $(\bar{\alpha}_1, \alpha_2, \alpha_3)$ and $(\alpha_1, \bar{\alpha}_2, \alpha_3)$; all vertices in row 1 in any copy of G labelled $(\alpha_1, \alpha_2, \alpha_3) \in \{0,1\}^3$ are adjacent to the vertices of the same name in the copies of G labelled $(\alpha_1, \bar{\alpha}_2, \alpha_3)$ and $(\alpha_1, \alpha_2, \bar{\alpha}_3)$; all vertices in row 2 in any copy of G labelled $(\alpha_1, \alpha_2, \alpha_3) \in \{0,1\}^3$ are adjacent to the vertices of the same name in the copies of G labelled $(\alpha_1, \alpha_2, \bar{\alpha}_3)$ and $(\bar{\alpha}_1, \alpha_2, \alpha_3)$; all vertices in row 4 in any copy of G labelled $(\alpha_1, \alpha_2, \alpha_3) \in \{0,1\}^3$ are adjacent to the vertices of the same name in the copies of G labelled $(\bar{\alpha}_1, \alpha_2, \alpha_3)$ and $(\alpha_1, \bar{\alpha}_2, \alpha_3)$; and so on. A few of the $\mathcal{Q}$-edges can be visualized as in Fig. 5(b) where $\mathcal{Q}$-edges across the first (resp. second, third) 'dimension' are depicted as running front-to-back (resp. right-to-left, top-to-bottom). In so far as we are aware, such a graph S has not featured before in the literature but it demonstrates the variety of graphs that can be defined within our framework. More on the pruning of interconnection networks using the semidirect product can be found in [38].

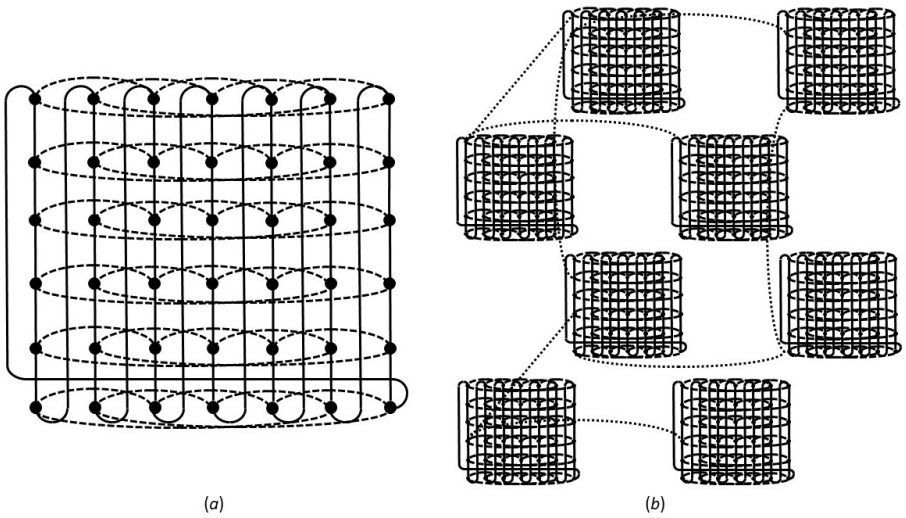

Figure 5: A particular graph G and some $\mathcal{Q}$-edges within S.

As our final illustration of the graphs that can be defined within our framework, let us involve the group $\mathcal{H}$ which has hitherto not featured. Define $\mathcal{G}$ by choosing: $r \geq 3$ and $c = 1$; $I_r = \emptyset$; and $\mathcal{H} = (\mathbb{Z}_r)^{p-1}$, for some $p \geq 2$, with $\Gamma_{\mathcal{H}} = \{\mathbf{e}_i, (r-1)\mathbf{e}_i : 1 \leq i \leq p-1\}$ (of course, $\mathbf{e}_i$ is the $(p-1)$-tuple with a 1 as the ith component and 0s elsewhere). So, G is an r-ary p-cube Q_p^r. Define $\mathcal{Q}$ by choosing: $m = 1$; $b_1 = 2$; $n_1 = q$, with $q \geq 1$; $d_1 = 1$; and M as the $q \times q$ matrix

which shifts each component of a vector 1 place to the right, with 'wrap-around' (similarly to the matrix M defined above for cube-connected cycles). So, the resulting graph G consists of 2^q copies of an r-ary p-cube Q_p^r where the $\mathcal{Q}$-edges are analogous to the $\mathcal{Q}$-edges of a cube-connected cycles. In so far as we are aware, such a graph S has not featured before in the literature. However, our construction shows that the 'core' graph G of a graph S within our framework need not necessarily be circulants.

There exist other interconnection networks that have been defined using group semidirect products but that do not fit into our framework. We now provide three examples.

Wrapped butterflies. Suppose that we define $\mathcal{G} = \mathbb{Z}_n = \langle \gamma \rangle$, for some $n \geq 3$, and $\mathcal{Q} = (\mathbb{Z}_2)^n$. For $\mathbf{q} \in \mathcal{Q}$, define $\varphi_\gamma : \mathbf{q} = (a_1, a_2, a_3, \ldots, a_n) \mapsto (a_n, a_1, a_2, \ldots, a_{n-1})$. This yields an action of $\mathcal{G}$ on $\mathcal{Q}$ and so we can build the semidirect product $\mathcal{Q} \rtimes \mathcal{G}$ w.r.t. this action. Let us define the set of generators Γ_S of $\mathcal{Q} \rtimes \mathcal{G}$ as $\Gamma_S = \{(1_e, \gamma), (1_e, \gamma^{-1}), (\mathbf{e}_n, \gamma^{-1}), (\mathbf{e}_1, \gamma)\}$ (note that this set is inverse-closed). The resulting graph $Cay(\mathcal{Q} \rtimes \mathcal{G}; \Gamma_S)$ is the *wrapped butterfly network* WB_n (see, e.g., [21]). Wrapped butterflies almost fit within our framework except that the chosen set of generators Γ_S means that $\mathcal{Q}$-edges do not join vertices of the same name (in different cycles of length n).

Generalized cube-connected cycles. The cube-connected cycles mentioned earlier were extended to *generalized cube-connected cycles* GCC in [8]. These graphs were somewhat loosely defined and almost fit within our framework. Suppose that when building $\mathcal{G}$ we choose: $r \geq 2$ and $c = 1$; $I_R = \emptyset$; and $\mathcal{H}$ as the trivial group. Hence, G is a cycle of length $r \geq 3$ or an edge, if $r = 2$. With regard to $\mathcal{Q}$, we choose: $m = 1$; $b_1 = 2$; and $n_1 = n$. Consequently, $\mathcal{Q}$ is the direct product of r copies of $\mathbb{Z}_2$. In [8], M was allowed to be *any* non-singular $n \times n$ matrix over $\{0, 1\}$ (with addition modulo 2) and the set of generators of the semidirect product $\mathcal{Q} \rtimes \mathcal{G}$ was left as user-defined (so the parameter d_1 plays no role). While the class of generalized cube-connected cycles GCC does not fit precisely within our framework, such graphs are Cayley graphs of the group $\mathcal{Q} \rtimes \mathcal{G}$. It was acknowledged in [8] that ensuring the connectivity of a generalized cube-connected cycle is problematic.

Supertoroids. Finally, another class of interconnection networks that has been defined using semidirect products but does not fit within our framework is the class of *supertoroids* as defined in [14, 13] (see [37] for a more accessible account). Supertoroids are Cayley graphs of the semidirect product $\mathbb{Z}_{c^2 l} \rtimes \mathbb{Z}_{ck}$, where $c \geq 2$ and $k, l \geq 1$. The group $\langle \alpha \rangle = \mathbb{Z}_{ck}$ acts on $\langle \beta \rangle = \mathbb{Z}_{c^2 l}$ as follows: define the map $\varphi_{\alpha^i} : \beta^j \mapsto \beta^{(1-icl)j}$. Define the set of generators $\Gamma_S = \{(\alpha^\epsilon, 1_e), (1_e, \beta^\epsilon) : \epsilon = \pm 1\}$ and let S be the graph $Cay(\mathbb{Z}_{c^2 l} \rtimes \mathbb{Z}_{ck}; \Gamma_S)$. The graph S can be thought of as follows.

- S consists of $c^2 l$ copies of a cycle C_{ck}, of length ck, whose vertex set is $\{0, 1, \ldots, ck - 1\}$; moreover, these copies can be labelled $0, 1, \ldots, c^2 l - 1$.

- Every vertex of one of these copies of C_{ck} is adjacent to exactly 2 vertices of the same name in different copies of C_{ck} as follows (where all arithmetic is modulo c^2l):
  - vertex 0 of copy j is adjacent to vertex 0 in copies $j-1$ and $j+1$
  - vertex 1 of copy j is adjacent to vertex 1 in copies $j+(1-cl)$ and $j-(1-cl)$
  - vertex 2 of copy j is adjacent to vertex 2 in copies $j+(1-2cl)$ and $j-(1-2cl)$

    ...
  - vertex $c-1$ of copy j is adjacent to vertex $c-1$ in copies $j+(1-(c-1)cl) = j+(1+cl)$ and $j-(1-(c-1)cl) = j-(1+cl)$
  - vertex c of copy j is adjacent to vertex c in copies $j+1$ and $j-1$

  and so on.

It is not difficult to show that S is 4-regular. Note that supertoroids almost fit within our framework except that the action of $\mathbb{Z}_{ck}$ on $\mathbb{Z}_{c^2l}$ is not as is required in Section 3.2.

The main point of us defining the various classes of graphs in this section is to show that, first, our framework is broad and encompasses a wide range of graph classes relevant to the construction of interconnection networks and, second, there is scope for possibly extending our framework in future so as to capture other classes of graphs that have hitherto been considered as interconnection networks. We shall return to this latter point when we present our conclusions.

4. Connectivity

Having defined our framework, we begin with an examination of connectivity. There are many properties that we would prefer any interconnection network to have and these properties (some of which we mentioned in the Introduction) need to be investigated for our graphs. An obvious place to start is to mirror what was done: in [28] for cube-connected circulants, where (as well as an optimal shortest-path routing algorithm, which is the main focus of this paper) explicit formulae for diameters were derived along with embeddings into hypercubes (and *vice versa*); and in [29] for recursive cubes of rings where the Wiener index, the vertex-forwarding index, the edge-forwarding index and the bisection width were studied.

However, we can shed light on the connectivity of toroidal semidirect product graphs using existing results and we do this now. (Note that what follows also fills a gap in [28, 29] where the connectivity of cube-connected cycles or recursive cubes of rings was not mentioned.) We say that a set of generators of some group (that is closed under inverses and does not contain the identity) is *quasi-minimal* if subsets of these generators can be linearly ordered as $B_1, B_2, \ldots, B_n$, for $n \geq 1$, so that:

- each B_i consists of either a generator of order 2 or a pair of generators that are inverses, so that these subsets of generators are pairwise disjoint and contain all generators
- for each $i = 1, 2, \ldots, n-1$, the subgroup generated by the generators of $B_1 \cup B_2 \cup \ldots \cup B_i$ is a proper subgroup of the subgroup generated by the generators of $B_1 \cup B_2 \cup \ldots \cup B_i \cup B_{i+1}$,

and we refer to such an ordering as a *quasi-minimal ordering* of the generators. This concept arose in [2] and [4] although it is Alspach's consideration of it in [3] that we use here: in [3], Alspach proved that any Cayley graph where the generating set of the group is quasi-minimal necessarily has connectivity equal to its degree, unless the Cayley graph is from an exceptional family in which case the connectivity is the degree minus 1. This exceptional family consists of Cayley graphs for which the quasi-minimal set of generators, linearly ordered via the subsets $B_1, B_2, \ldots, B_n$, for $n \geq 2$, is such that: B_1 consists of a generator of order 2; and each B_i, for $2 \leq i \leq n$, consists of a pair of (inverse) generators of order 4 so that each of these generators commutes with the generator in B_1 and is such that its square is equal to the generator in B_1. If a regular graph has connectivity equal to its degree then we say that it is *maximally connected*.

With regard to the generators of a toroidal semidirect product graph $S = Cay(\mathcal{Q} \rtimes \mathcal{G}; (1_e \times \Gamma_{\mathcal{G}}) \cup (\Gamma_{\mathcal{Q}} \times 1_e))$, we can proceed as follows.

- We can arbitrarily order the generators of $\Gamma_{\mathcal{Q}} \times 1_e$ via sets $B_1, B_2, \ldots B_d$ of pairs of inverses or elements of order 2, as is appropriate, so that we have a quasi-minimal ordering.
- If the set of generators $\Gamma_{\mathcal{G}}$ is a quasi-minimal set of generators of $\mathcal{G}$ then we can extend $B_1, B_2, \ldots B_d$ to a quasi-minimal ordering of the generators of $(1_e \times \Gamma_{\mathcal{G}}) \cup (\Gamma_{\mathcal{Q}} \times 1_e)$.

Given that no toroidal semidirect product graph is a member of the above exceptional family, Alspach's result immediately gives us the following corollary.

Corollary 1. *The toroidal semidirect product graph* $S = Cay(\mathcal{Q} \rtimes \mathcal{G}; (1_e \times \Gamma_{\mathcal{G}}) \cup (\Gamma_{\mathcal{Q}} \times 1_e))$ *is maximally connected if there is a quasi-minimal ordering of the generators of* $\Gamma_{\mathcal{G}}$.

We immediately obtain the following result.

Corollary 2. *The cube-connected circulants, and so the recursive cubes of rings, are maximally connected.*

5. A framework for shortest paths in S

Our aim is now to develop an efficient algorithm to find a shortest path from any given source vertex *start* of S to any given target vertex *end* (where S is the interconnection network defined in Section 3.3). As S is a Cayley graph,

so it is vertex-transitive; consequently, w.l.o.g., we may assume that our source vertex *start* is $(1_e, (1_e, 1_e)) = (1_e, (1_e, 0, 0))$ in S. We begin by taking a less algebraic perspective of S and then go on to prove that finding a shortest path from *start* to *end* in S is equivalent to finding a certain type of 'covering' walk, called a D-walk, in a copy of G_0 (recall that G_0 was defined in Section 3.1 and there are many copies of G_0 within S, as was noted in Section 3.4).

5.1. The torus T associated with S

For any vertex $(\mathbf{q}, (\sigma, \gamma^i))$ of S and for each generator $\mathbf{q}^{\epsilon}_{k,j} \in \Gamma_{\mathcal{Q}}$, we have that $(\mathbf{q}, (\sigma, \gamma^i))(\mathbf{q}^{\epsilon}_{k,j}, (1_e, 1_e)) = (\mathbf{q} + \mathbf{q}^{\epsilon}_{k,j'}, (\sigma, \gamma^i))$, where $M^i \mathbf{q}^{\epsilon}_{k,j} = \mathbf{q}^{\epsilon}_{k,j'}$ (we use additive notation as we are regarding elements of $\mathcal{Q}$ as n-tuples of integers). Consequently, we can think of $\mathcal{Q}$-edges incident with $(\mathbf{q}, (\sigma, \gamma^i))$ in S as corresponding to positive and negative 'moves' over d specific 'dimensions' within the

$$\overbrace{b_1 \times \ldots \times b_1}^{n_1 \text{ times}} \times \overbrace{b_2 \times \ldots \times b_2}^{n_2 \text{ times}} \times \ldots \times \overbrace{b_m \times \ldots \times b_m}^{n_m \text{ times}}\text{-torus}$$

T of dimension n associated with the group $\mathcal{Q}$; more precisely, T is the Cayley graph $Cay(\mathcal{Q}; \Gamma^*_{\mathcal{Q}})$, where $\Gamma^*_{\mathcal{Q}} = \{\mathbf{q}^{\epsilon}_{k,j} : 1 \leq k \leq m; 1 \leq j \leq n_k; \epsilon = \pm 1\}$ (we remove duplications of the form $\mathbf{q}^{-1}_{m,j}$ from $\Gamma^*_{\mathcal{Q}}$ if $b_m = 2$ and think of hypercubes as tori). We call these specific d dimensions of T the dimensions *covered* by the vertex $(\mathbf{q}, (\sigma, \gamma^i))$. We reiterate that, by construction, if a dimension of T is covered by some vertex of S then we can 'move' positively or negatively across the $\mathcal{Q}$-edge in this dimension within S, from one copy of G to another (we can think of the copies of G within S as being labelled by elements of $\mathcal{Q}$ or by vertices of T). What the definition of S does, via the action of $\mathcal{G}$ on $\mathcal{Q}$, is to vary the dimensions of T over which we might move, from any chosen vertex of S. Compare this with the direct product $G \times T$ where from any chosen vertex, we can move over *any* dimension of T. So, our construction reduces the number of edges in S in comparison with the number of edges in $G \times T$; of course, we wish to do this but so that we do not harm the efficiency of S as an interconnection network. As we stated earlier, it is important to work with interconnection networks of low and constant degree.

5.2. Obtaining a canonical edge-ordering

We now show that any shortest path from vertex *start* to vertex *end* in S has a particular canonical form.

Lemma 3. *Let ρ be a shortest path from vertex start to vertex end in S. We may assume that ρ consists of a prefix ρ_0 of column- and $\mathcal{Q}$-edges followed by a sequence ρ_1 of row-edges followed by a suffix ρ_2 of $\mathcal{H}$-edges.*

Proof. Let ρ be a shortest path from vertex *start* to vertex *end* in S. The path ρ consists of a sequence of column-, row-, $\mathcal{H}$- and $\mathcal{Q}$-edges. Let e be an $\mathcal{H}$-edge in ρ, corresponding to some generator of $\mathcal{Q} \rtimes \mathcal{G}$ of the form $(1_e, (\sigma, 1_e))$, and suppose that e is followed in ρ by a row-edge or a column-edge e', corresponding to some

generator of the form $(1_e, (1_e, \gamma^k))$. As $(1_e, (\sigma, 1_e))(1_e, (1_e, \gamma^k)) = (1_e, (\sigma, \gamma^k)) = (1_e, (1_e, \gamma^k))(1_e, (\sigma, 1_e))$, we can replace e and e' in ρ by a row-edge or a column-edge, respectively, followed by an $\mathcal{H}$-edge. Suppose now that the $\mathcal{H}$-edge e is followed in ρ by a $\mathcal{Q}$-edge, corresponding to some generator of the form $(\mathbf{q}, (1_e, 1_e))$. We have that $(1_e, (\sigma, 1_e))(\mathbf{q}, (1_e, 1_e)) = (\mathbf{q}, (\sigma, 1_e)) = (\mathbf{q}, (1_e, 1_e))(1_e, (\sigma, 1_e))$ and so we can replace e and e' in ρ by a $\mathcal{Q}$-edge followed by a $\mathcal{H}$-edge. Hence, w.l.o.g., we may assume that all $\mathcal{H}$-edges of ρ appear as a suffix.

Let e be a row-edge, corresponding to a generator of the form $(1_e, (1_e, \gamma^{kr}))$, and suppose that e is followed in ρ by a $\mathcal{Q}$-edge e', corresponding to some generator of the form $(\mathbf{q}, (1_e, 1_e))$. We have that $(1_e, (1_e, \gamma^{kr}))(\mathbf{q}, (1_e, 1_e)) = (M^{kr}\mathbf{q}, (1_e, \gamma^{kr})) = (\mathbf{q}, (1_e, \gamma^{kr})) = (\mathbf{q}, (1_e, 1_e))(1_e, (1_e, \gamma^{kr}))$ and so we can replace e and e' in ρ with a $\mathcal{Q}$-edge followed by a row-edge. It is trivial to see that we can replace a row-edge followed by a column-edge in ρ with a column-edge followed by a row-edge. Hence, w.l.o.g., we may assume that all $\mathcal{H}$-edges of ρ appear as a suffix and that all row-edges of ρ appear immediately prior to the suffix of $\mathcal{H}$-edges. The result follows. □

Henceforth, we assume that any shortest path ρ in S from vertex *start* to vertex *end* has the form $\rho_0\rho_1\rho_2$ given by Lemma 3.

The graph S consists of $|\mathcal{Q}|$ copies of G, labelled by the elements $\mathbf{q} \in \mathcal{Q}$, and in each of these copies of G, there are $|\mathcal{H}|$ copies of G_0. In the copy of G labelled $1_e \in \mathcal{Q}$, there is a copy of G_0 containing *start*; call this copy $G_{0,s}^{1_e}$. Let $G_{0,s}^{\mathbf{q}}$ be the copy of G_0 in the copy of G labelled $\mathbf{q} \in \mathcal{Q}$ that corresponds to $G_{0,s}^{1_e}$, and let K_0 be the subgraph of S induced by the vertices of all of these copies $G_{0,s}^{\mathbf{q}}$; that is, K_0 is the subgraph of S induced by the vertices of $\{(\mathbf{q}, (1_e, \gamma^i)) : \mathbf{q} \in \mathcal{Q}; 0 \leq i \leq cr - 1\}$. An alternative definition of K_0 is as the closure of the subgraph $G_{0,s}^{1_e}$ by traversing column-, row- and $\mathcal{Q}$-edges. Note that the prefix $\rho_0\rho_1$ of any canonical shortest path ρ consists of vertices in K_0 and let *end*$'$ be the vertex reached after traversing the sequence of edges $\rho_0\rho_1$. In particular, if $end = (\mathbf{q}, (\sigma, \gamma^j))$ then $end' = (\mathbf{q}, (1_e, \gamma^j))$. Hence, the suffix ρ_2 is isomorphic to a shortest path in $Cay(\mathcal{H}; \Gamma_{\mathcal{H}})$ from 1_e to σ. Consequently, it remains to find a shortest path from $start = (1_e, (1_e, 1_e))$ to $end' = (\mathbf{q}, (1_e, \gamma^j))$ in K_0.

Project K_0 onto a copy of G_0, which we call $\bar{G}_0$, via $(\mathbf{q}, (1_e, \gamma^i)) \mapsto (x, y)$, where $\mathbf{q} \in \mathcal{Q}$, $0 \leq i \leq cr - 1$ and $i = ry + x$, with $0 \leq x \leq r - 1$ and $0 \leq y \leq c - 1$. Note that any two vertices of K_0 that are projected onto the same vertex of $\bar{G}_0$ cover the same dimensions; hence, it makes sense to talk about the dimensions covered by vertices in $\bar{G}_0$. Finding a path $\rho_0\rho_1$ in K_0 from *start* to $end' = (\mathbf{q}, (1_e, \gamma^j))$, where $j = y_0c + x_0$ with $0 \leq x_0 \leq r - 1$ and $0 \leq y_0 \leq c - 1$, is equivalent to finding a walk π in $\bar{G}_0$ from $(0, 0)$ to (x_0, y_0) so that:

- π consists of a prefix π_0 of column-edges followed by a suffix π_1 of row-edges
- the set of dimensions of the torus T associated with $\mathcal{Q}$ covered by the vertices on the walk π_0 suffices to build a path from 1_e to $\mathbf{q}$ in T; that

is, if D is the set of dimensions for which the corresponding component of $\mathbf{q}$ is non-zero (we are thinking of $\mathbf{q}$ as a tuple of integers) then the set of dimensions covered by the vertices of π_0 contains D.

A walk π, as above, can be 'expanded' to a path within K_0 from $(1_e, (1_e, 0, 0))$ to $end' = (\mathbf{q}, (1_e, x_0, y_0))$ by enmeshing $\mathcal{Q}$-edges amongst the column-edges of π_0, at the appropriate point in the expansion, and then to conclude with the suffix π_1 of row-edges to get to end'. Note that every dimension from D is necessarily involved in a move on any path within K_0 from $(1_e, (1_e, 0, 0))$ to $(\mathbf{q}, (1_e, x_0, y_0))$. Consequently, in order to find our shortest path $\rho_0\rho_1$ from *start* to end' in K_0, we require a shortest walk $\pi = \pi_0\pi_1$ in $\bar{G}_0$ from $(0, 0)$ to (x_0, y_0) consisting of a walk of column-edges π_0 followed by a path of row-edges π_1 so that every dimension in D is covered by some vertex on the walk π_0; that is, π_0 and π are what we define as *D-walks*.

Remark 4. *It can be appreciated now as to why within our framework we have restricted $\mathcal{Q}$ to be a direct product of cyclic groups and $\Gamma_{\mathcal{Q}}$ to consist of some 'generators' of these individual groups: shortest-path routing in the torus T is governed solely by the dimensions for which $\mathbf{q}$ is non-zero and so we need only look for a shortest D-walk π_0 in $\bar{G}_0$, safe in the knowledge that once found we can extend it to a shortest path in K_0 from $(1_e, (1_e, 0, 0))$ to $(\mathbf{q}, (1_e, x_0, y_0))$. If $Cay(\mathcal{Q}; \Gamma^*_{\mathcal{Q}})$, for some set of elements $\Gamma^*_{\mathcal{Q}}$, were to be a more complex, 'non-dimensional' graph then, in general, we cannot simply worry about exploring walks in $\bar{G}_0$ that cover an appropriate set of generators. In particular, there might be various walks π_0 that cover various sets of generators, each set of which enables us to build a path in K_0 from $(1_e, (1_e, 0, 0))$ to $(\mathbf{q}, (1_e, x_0, y_0))$, but where it is not clear which walk π_0 and which set of generators yields a shortest such path.*

6. Types of walks

From the previous section, we have reduced our task to finding a shortest walk $\pi = \pi_0\pi_1$ in $\bar{G}_0$ from $(0, 0)$ to (x_0, y_0) so that π_0 is a D-walk of column-edges and π_1 is a path of row-edges. There are two essential types of walks from which the shortest one is drawn:

- Type A: $c = 1$ or $c > 1$ and the D-walk π_0 does not pass through either $(0, 1)$ or $(0, c - 1)$
- Type B: $c > 1$ and the D-walk π_0 passes through either $(0, 1)$ or $(0, c-1)$.

We shall deal with these two types separately.

6.1. Walks of Type A

There are two sub-types of Type A walks.

Lemma 5. *Let $\pi = \pi_0\pi_1$ be a shortest walk in $\bar{G}_0$ from $(0, 0)$ to (x_0, y_0) of Type A. This walk π is of one of two sub-types:*

- *Type A(*i*): the D-walk π_0 is from $(0,0)$ to $(x_0,0)$*
- *Type A(*ii*): the D-walk π_0 is from $(0,0)$ to $(x_0,c-1)$.*

PROOF. With reference to Fig. 1, the D-walk π_0 must end at a vertex on the path of column-edges from $(1,c-1)$ to $(r-1,0)$ through $(0,0)$. □

Of course, in Lemma 5, if $c=1$ then there is only one sub-type. Also, if $I_R=\emptyset$ then for a Type A(i) (resp. Type A(ii)) walk to exist we need that $y_0=0$ (resp. $y_0=c-1$), and we have that π_1 is the empty walk (see Section 3.1 for a reminder of the definition of I_R).

The following lemmas detail upper bounds on the lengths of the walk π_0 in Lemma 5. However, within their actual proofs there is additional information in the form of explicit descriptions of π_0, for the various cases that can occur (the hypotheses of Lemmas 6–10 cover all possibilities for a Type A D-walk). We will use these explicit descriptions when we develop algorithms in Section 6.2. We do not provide these descriptions of π_0 in the statements of the lemmas as to do so would unnecessarily complicate these statements, which provide succinct bounds on the length of π_0 and culminate in Corollary 11; the descriptions of π_0 can be somewhat technically involved (but are readily obtained from the proofs when we come to develop our algorithms).

We remind the reader that the parameters μ and M are defined in Section 3.2 (r and c are core parameters in the definition of the group $\mathcal{G}$ in Section 3.1 and n is a core parameter in the definition of the group $\mathcal{Q}$ in Section 3.2).

Lemma 6. *Let π_0 be a shortest D-walk in $\bar{G}_0$ from $(0,0)$ to $(x_0,0)$, of Type A(i), and suppose further that $c>1$ and $0\le x_0\le\mu$. The length of π_0 is at most $2\mu-x_0$.*

PROOF. The location of vertex $(0,0)$ in $\bar{G}_0$ can be visualized as in Fig. 6 (though it is feasible that $(0,1)$ and $(0,c-1)$ are one and the same vertex, i.e., when $c=2$). The different batches of (the n) dimensions of T covered by each vertex depicted are D_i, for $i\in\{0,\pm1,\pm2,\ldots,\pm(r-1)\}$. Note that all n dimensions are covered by the shaded vertices (namely $\{(i,0):0\le i\le\mu\}$) as well as by the outlined vertices (namely $\{(0,0)\}\cup\{(i,c-1):r-\mu\le i\le r-1\}$). Note also that these batches of dimensions repeat every r vertices (as $|M|$ divides r).

Our D-walk π_0 will have length at most $2\mu-x_0$, as the D-walk π_0' defined as

$$(0,0),(1,0),\ldots,(\mu'-1,0),(\mu',0),(\mu'-1,0),\ldots,(x_0,0)$$

has length at most $2\mu-x_0$, where μ' is the least number from $\{x_0,x_0+1,\ldots,\mu\}$ so that the vertices of the walk cover all dimensions of D. Of course, there may be a shorter appropriate D-walk than π_0' (although the 'anti-clockwise' D-walk consisting of the path from $(0,0)$ through $(r-1,c-1)$ and on to $(x_0,0)$, of length $cr-x_0$, is not shorter).

The question is: 'What does a shortest appropriate D-walk π_0 look like?' In theory, π_0 might wander from $(0,0)$ to $(x_0,0)$, changing direction many times. However, we can see that there are limits on these changes of direction. Suppose, for example, that, with reference to Fig. 6, the walk π_0:

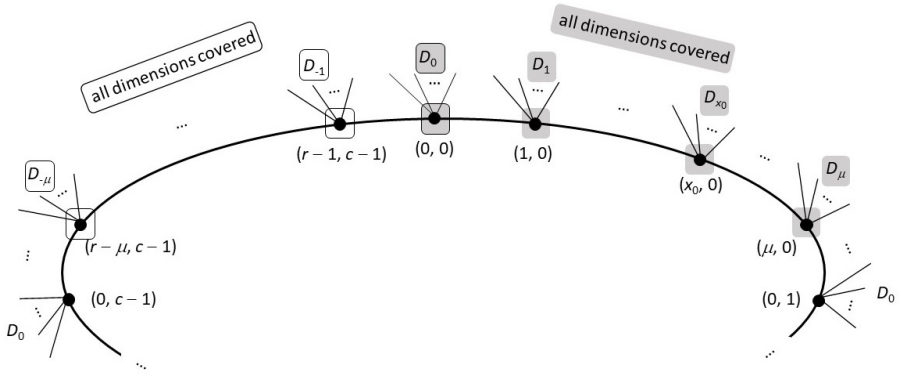

Figure 6: The locality of $(0,0)$ in $\bar{G}_0$.

- starts by moving 'clockwise' from $(0,0)$ (to $(1,0)$) and later changes direction to return to $(0,0)$ (this is the walk τ_1);
- then moves 'anti-clockwise' from $(0,0)$ (to $(r-1,c-1)$) and later changes direction to return to $(0,0)$ (this is the walk τ_2);
- then moves 'clockwise' from $(0,0)$ to $(x_0,0)$ (this is the walk τ_3);

so, $\pi_0 = \tau_1\tau_2\tau_3$. As π_0 is a shortest appropriate D-walk, τ_1 'turns' at some vertex $(t,0)$ with $t \leq \mu$ (as otherwise τ_1 would pass through $(x_0,0)$ and we could truncate it 'on the way back' after the turn to obtain a shorter appropriate D-walk). If $t \leq x_0$ then the walk $\tau_2\tau_3$ is a shorter appropriate D-walk; and if $x_0 < t$ then $\tau_2\tau_1'$, where τ_1' is τ_1 truncated at $(x_0,0)$ 'on the way back' after the turn, is a shorter appropriate D-walk. By undertaking a simple case-by-case analysis, it is not difficult to see that our shortest D-walk is either π_0' or one of the form

$$(0,0),(r-1,c-1),\ldots,(\mu_l+1,c-1),(\mu_l,c-1),(\mu_l+1,c-1),\ldots$$
$$\ldots,(0,0),\ldots,(\mu_r-1,0),(\mu_r,0),(\mu_r-1,0),\ldots,(x_0,0)$$

where: $r-1 \geq \mu_l \geq r-\mu$; $(\mu_l, c-1)$ covers some dimension of D; and μ_r is the least number from $\{x_0, x_0+1, \ldots, \mu\}$ so that all dimensions of D are covered by the vertices of this walk. We call such a walk the *μ_l-left D-walk* (note that given D and μ_l, the walk is thereafter prescribed). The 'shape' of the D-walk π_0', in relation to $\bar{G}_0$, can be visualized as a dotted line in Fig. 7(a), whereas the 'shape' of a μ_l-left D-walk can be visualized as in Fig. 7(b). Collectively, we call the D-walk π_0' together with all the μ_l-left D-walks the set of *left D-walks* (from $(0,0)$ to $(x_0,0)$). Our shortest D-walk will have length at most $2\mu - x_0$.□

Lemma 7. *Let π_0 be a shortest D-walk in $\bar{G}_0$ from $(0,0)$ to $(x_0, c-1)$, of Type A(ii), and suppose further that $c > 1$ and $r - \mu \leq x_0 \leq r-1$. The length of π_0 is at most $2\mu - (r - x_0)$.*

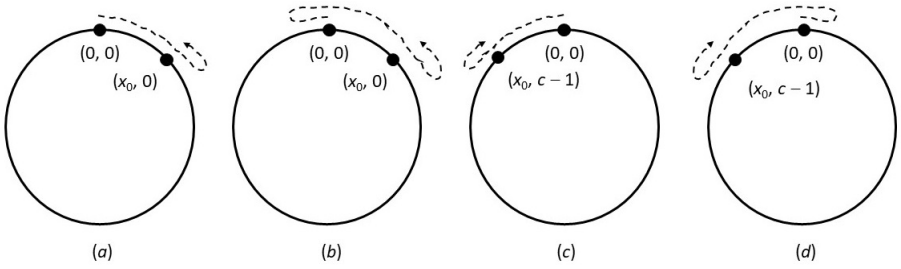

Figure 7: The 'shapes' of our walks relative to $\bar{G}_0$.

PROOF. By proceeding similarly to the proof of Lemma 6, we obtain that: our new D-walk π'_0 is the walk of length at most $2\mu - (r - x_0)$ that is analogous to the D-walk π'_0 from the proof of Lemma 6 (but heads off in an 'anti-clockwise' direction; see Fig. 7(c)); and our *μ_r-right D-walks*, where $1 \leq \mu_r \leq \mu$ so that $(\mu_r, 0)$ covers some dimension of D, are defined analogously to the μ_l-left D-walks from the proof of Lemma 6 (a μ_r-right D-walk starts off in a 'clockwise' direction; see Fig. 7(d)). Collectively, we call the D-walk π'_0 together with all the μ_r-right D-walks the set of *right D-walks* (from $(0,0)$ to $(x_0, c-1)$). Our shortest D-walk will have length at most $2\mu - (r - x_0)$. □

Lemma 8. *Let π_0 be a shortest D-walk in $\bar{G}_0$ from $(0,0)$ to $(x_0,0)$, of Type A(i), and suppose further that $c > 1$ and $\mu < x_0 \leq r - 1$. The length of π_0 is x_0.*

PROOF. It must be the case that π_0 is the 'clockwise' path from $(0,0)$ to $(x_0,0)$ of length x_0 as all dimensions are necessarily covered. □

Lemma 9. *Let π_0 be a shortest D-walk in $\bar{G}_0$ from $(0,0)$ to $(x_0, c-1)$, of Type A(ii), and suppose further that $c > 1$ and $1 \leq x_0 < r - \mu$. The length of π_0 is $r - x_0$.*

PROOF. It must be the case that π_0 is the 'anti-clockwise' path from $(0,0)$ to $(x_0, c-1)$ of length x_0 as all dimensions are necessarily covered. □

Lemma 10. *Let π_0 be a shortest D-walk in $\bar{G}_0$ from $(0,0)$ to $(x_0,0)$ when $c = 1$. The length of π_0 is at most $\max\{2\mu, r\} \leq 2r - 2$.*

PROOF. We have that $\mu < |M|$ and $|M|$ divides r. Suppose, in the first instance, that $|M| \neq r$; so, $\mu < \frac{r}{2}$ and we have that $r - \mu > \mu$. If $0 \leq x_0 \leq \mu$ then we are in the situation of Lemma 6 (where the length of π_0 is at most $2\mu - x_0$; notice that the 'anti-clockwise' path from $(0,0)$ to $(x_0,0)$ has length greater than $2\mu - x_0$ whereas the D-walk π_0 from Lemma 6 has length at most $2\mu - x_0$). Similarly, if $r - \mu \leq x_0 \leq r - 1$ then we are in the situation of Lemma 7 (when the length of π_0 is at most $2\mu - (r - x_0)$). Finally, if $r - \mu > x_0 > \mu$ then the D-walk π_0 is either the 'clockwise' or the 'anti-clockwise' path from $(0,0)$ to $(x_0,0)$ of length

x_0 or $r - x_0$, respectively, depending upon which is shortest (as all dimensions are necessarily covered by both of these walks).

Suppose now that $|M| = r$. If $r - \mu > \mu$ then the reasoning in the previous paragraph applies; hence, we may suppose that $2\mu \geq r$. In this case, if $x_0 \neq 0$ then we take π_0 to be the shortest D-walk from the left D-walks (as defined in the proof of Lemma 6) and the right D-walks (as defined in the proof of Lemma 7); so, our D-walk π_0 has length at most $\min\{2\mu - x_0, 2\mu - (r - x_0)\}$. Alternatively, if $x_0 = 0$ then we take π_0 to be the shortest D-walk from the left D-walks (as defined in the proof of Lemma 6), the right D-walks (as defined in the proof of Lemma 7) and the 'anti-clockwise' full circular walk from $(0, 0)$, through $(0, r - 1)$ and on to $(0, 0)$; so, our D-walk π_0 has length at most r. Let us remark that if $c = 1$ and $r = 2$ then strictly speaking, $\bar{G}_0$ is not a cycle but an edge. However, if we think of this edge as a cycle of length 2 then the reasoning above goes through (albeit much more simply). □

By examining the lengths of walks described in the various lemmas above, we obtain the following result.

Corollary 11. *Let π_0 be a shortest D-walk in $\bar{G}_0$ from $(0, 0)$ to: $(x_0, 0)$, if π_0 is of Type A(i); or to $(x_0, c - 1)$, if π_0 is of Type A(ii). The length of π_0 is at most* $\max\{2\mu, r\}$.

Having secured our shortest D-walk π_0, the path π_1 is simply isomorphic to the shortest path in the circulant graph R (see Section 3.1) either from 0 to y_0 or from $c - 1$ to y_0, as appropriate, if these paths exist in R (for we have made no assumptions about the connectivity of R).

6.2. Algorithms for D-walks of Type A

We can now convert the descriptions of the walk π_0 in $\bar{G}_0$, in the various lemmas of Section 6.1, into an algorithm for its construction (assuming it exists). Suppose that we are given a subset of dimensions D and that $0 \leq x_0 \leq \mu$ and $c > 1$, so that we are looking for a shortest D-walk from $(0, 0)$ to $(x_0, 0)$ (hence, we are in the situation of Lemma 6). It is straightforward to design an algorithm, call it *ShortestLeftDWalk*, that outputs a shortest D-walk from $(0, 0)$ to $(x_0, 0)$ from the set of left D-walks; this algorithm *ShortestLeftDWalk* is detailed as Algorithm 1.

The first for-loop in lines 1–4 assigns the various batches of dimensions to the vertices $(\mu, c-1), (\mu+1, c-1), \ldots, (r-1, c-1), (0, 0), (1, 0), \ldots, (\mu-1, 0), (\mu, 0)$ which we think of as i running through $-\mu, -(\mu - 1), \ldots, -1, 0, 1, \ldots, \mu - 1, \mu$. This can be undertaken in $O(n\mu)$ time (note that although the construction of a set of dimensions in lines 2–3 is phrased in terms of matrix multiplication, we can use permutations to obtain a more efficient implementation). The second for-loop in lines 5–7 recomputes D_{x_0} as all those dimensions covered by vertices $(0, 0), (1, 0), \ldots, (x_0, 0)$. This can be undertaken in $O(n\mu)$ time. The third for-loop in lines 8–23 accounts for when the vertex μ_l of a left D-walk (as defined in the proof of Lemma 6) is chosen to be each vertex of $(r - 1, c - 1), (r - 2, c -$

Algorithm 1 *ShortestLeftDWalk*

for $i = -\mu$ up to μ **do**
  $D_i = \{l : M^i \mathbf{e}_j = \mathbf{e}_l$ where $1 \le j \le d_1, n_1 + 1 \le j \le n_1 + d_2, \ldots$
    $\ldots, n_1 + \ldots + n_{m-1} + 1 \le j \le n_1 + \ldots + n_{m-1} + d_m\}$
end for
for $i = 0$ to $x_0 - 1$ **do**
  $D_{x_0} = D_{x_0} \cup D_i$
end for
for $i = -1$ down to $-\mu$ **do**
  if $i \neq -1$ **then**
    $D_i = D_i \cup D_{i+1}$
  end if
  $\bar{D}_i = D \setminus (D_i \cup D_{x_0})$
  if $\bar{D}_i = \emptyset$ **then**
    $\mu_r^i = x_0$
  else
    $\mu_r^i = x_0$
    while $\bar{D}_i \neq \emptyset$ **do**
      $\mu_r^i = \mu_r^i + 1$
      $\bar{D}_i = \bar{D}_i \setminus D_{\mu_r^i}$
    end while
  end if
  $length_i = 2|i| + x_0 + 2(\mu_r^i - x_0)$
end for
$\bar{D}_0 = D \setminus D_{x_0}$
if $\bar{D}_0 = \emptyset$ **then**
  $\mu_r^0 = x_0$
else
  $\mu_r^0 = x_0$
  while $\bar{D}_0 \neq \emptyset$ **do**
    $\mu_r^0 = \mu_r^0 + 1$
    $\bar{D}_0 = \bar{D}_0 \setminus D_{\mu_r^0}$
  end while
end if
$length_0 = x_0 + 2(\mu_r^0 - x_0)$
let j be s.t. $length_j$ is minimum from $\{length_j : j = 0, -1, \ldots, -\mu\}$
return $(j, length_j, \mu_r^j)$

$1), \ldots, (r-\mu, c-1)$; that is, when i is $-1, -2, \ldots, -\mu$. Within this for-loop, code is as follows.

- The first if-statement in lines 9–11 ensures that D_i is recomputed as all the dimensions covered by vertices $(r+i, c-1), (r+i+1, c-1), \ldots, (r-1, c-1)$. This can be undertaken in $O(n)$ time.
- The next statement in line 12 details the dimensions $\bar{D}_i$ remaining to be covered if μ_l is set as $(r+i, c-1)$, as all dimensions of D covered by $(r+i, c-1), (r+i+1, c-1), \ldots, (0,0), \ldots, (x_0-1, 0), (x_0, 0)$ are catered for. This can be undertaken in $O(n)$ time.
- The next if-statement in lines 13–21 computes the value μ_r (as defined in the proof of Lemma 6) for the current left D-walk; we denote this value by μ_r^i. This can be undertaken in $O(n(r-\mu))$ time.
- We compute the length $length_i$ of the current D-walk in line 22. This can be undertaken in $O(1)$ time.

So, the for-loop in lines 8–23 can be undertaken in $O(n(r-\mu)\mu)$ time. Finally, in lines 24–34 we compute the length of the path π_0', which can be undertaken in $O(n(r-\mu))$ time, and then in line 35 the minimum length from all left D-walks, which can be undertaken in $O(\mu)$ time. Note that the actual minimum-length path can be trivially reconstructed from the pair (j, μ_r^j) output in line 36 in $O(\mu)$ time. Hence, the time complexity of *ShortestLeftDWalk* is $O(r^2 n)$.

Of course, if we are in the situation of Lemma 7 then there is an analogous algorithm *ShortestRightDWalk* (of the same time complexity) that outputs a shortest D-walk in $\bar{G}_0$ from $(0,0)$ to $(x_0, c-1)$ from the set of right D-walks (as defined in the proof of Lemma 7) in the case that $r-\mu \leq x_0 \leq r-1$ and $c > 1$. If we are in the situation of Lemma 8 or Lemma 9 then we can trivially output the corresponding D-walk with time complexity $O(r)$. Similarly, if we are in the situation of Lemma 10 then we can use the existing algorithms (together with a check with regard to the 'anti-clockwise' full circular walk in the case when $x_0 = 0$) to obtain a shortest D-walk as required, again with time complexity $O(r^2 n)$. Consequently, there is an algorithm, call it *ShortestDWalk*, of time complexity $O(r^2 n)$ that given an additional input value that is either 0 or $c-1$, computes a shortest D-walk from $(0,0)$ to either $(x_0, 0)$ or $(x_0, c-1)$, respectively, in $\bar{G}_0$ that does not pass through either $(0,1)$ or $(0, c-1)$. By Corollary 11, such a D-walk has length at most $2r-2$.

Concerning computing the shortest path π_1 in the circulant graph R (if it exists), we can simply use a breadth-first search to undertake this in $O(c|I_R|)$ time. It might be thought that there could be scope for a more efficient shortest-path algorithm in R by representing R as the binary representations of c and the elements of I_R, rather than as a $c \times c$ adjacency matrix. However, it was shown in [5] that it is **NP**-hard to compute shortest-paths in circulant graphs when the graph is input via this concise representation. Of course, the length of the path π_1 is at most the diameter of R, namely $diam(R)$. So, in summary, we obtain the following result.

Corollary 12. *If there is a Type A D-walk in $\bar{G}_0$ from $(0,0)$ to (x_0,y_0) then this walk has length at most $2r-2+diam(R)$ and we can find and output this walk in $O(r^2n+c|I_R|)$ time.*

6.3. D-Walks of Type B

Let us now consider finding a shortest walk $\pi=\pi_0\pi_1$ from $(0,0)$ to (x_0,y_0) in $\bar{G}_0$ so that π_0 is a D-walk of column-edges that passes through either $(0,1)$ or $(0,c-1)$, with $c>1$, and π_1 is a path of row-edges (such a walk always exists). We do this mindful of the walks of Type A described above; that is, if we can be sure that there is a shorter walk of Type A than a potential walk of Type B then there is no need to consider the walk of Type B.

6.3.1. The case when $I_R=\emptyset$

We dispense with the case when $I_R=\emptyset$ first; so, suppose that $I_R=\emptyset$. If $y_0=0$ then the shortest Type B walk from $(0,0)$ to $(x_0,0)$ that passes through either $(0,1)$ or $(0,c-1)$ has length $\min\{2r-x_0,cr-x_0\}$. However, the Type A D-walk obtained in either Lemma 6 or Lemma 8 in Section 6.1 above, has length at most $\max\{x_0,2\mu-x_0\}$, which is less than the length of our Type B walk. So, we may assume that $y_0\neq 0$. A similar argument yields that we may assume that $1\leq y_0\leq c-2$ or ($y_0=c-1$ and $x_0=0$). Hence, the shortest Type B walk π from $(0,0)$ to (x_0,y_0) is the 'clockwise' path of length y_0r+x_0 or the 'anti-clockwise' path of length $(c-y_0)r-x_0$, and so π has length at most $\frac{cr}{2}$.

6.3.2. Exploring Type B D-walks when $I_R\neq\emptyset$

Henceforth, we assume that $I_R\neq\emptyset$. Note that any walk π_0 from $(0,0)$ that passes through $(0,1)$ or $(0,c-1)$ is necessarily a D-walk.

We begin by ruling out some possible scenarios (given the Type A analysis from Section 6.1). Suppose that the D-walk π_0 ends at (x_0,i), where $0\leq i\leq c-1$ (in general there may not exist a path from vertex i of R to vertex y_0 as we have not insisted that R is connected). Suppose further that $i=0$. Any walk π_0 from $(0,0)$ to $(x_0,0)$ via $(0,1)$ or $(0,c-1)$ has length at least $2r-x_0$. However, the Type A D-walk obtained in either Lemma 6 or Lemma 8 has length at most $\max\{x_0,2\mu-x_0\}$, which is less than the length of π_0. So, we may assume that $i\neq 0$. An analogous argument holds when $i=c-1$ and $x_0>0$ (using Lemma 7 and Lemma 9). Thus we may assume that $1\leq i\leq c-2$ or ($i=c-1$ and $x_0=0$). With reference to Fig. 6, (x_0,i) is any vertex on $\bar{G}_0$ reachable by a path of column-edges starting at $(0,1)$ and moving towards but not past $(0,c-1)$. In particular, the shortest D-walk of column-edges from $(0,0)$ to (x_0,i) is either the 'clockwise' path through $(0,1)$ or the 'anti-clockwise' path through $(0,c-1)$.

There is also an anomalous case to deal with before we proceed. Suppose that $c=2$; so, $(x_0,i)=(0,1)$ and $I_R=\{1\}$. If $y_0=0$ then the shortest Type B walk π from $(0,0)$ to (x_0,y_0) has length $r+1$; and if $y_0=1$ then the shortest Type B walk π from $(0,0)$ to (x_0,y_0) has length r. The lengths of these walks

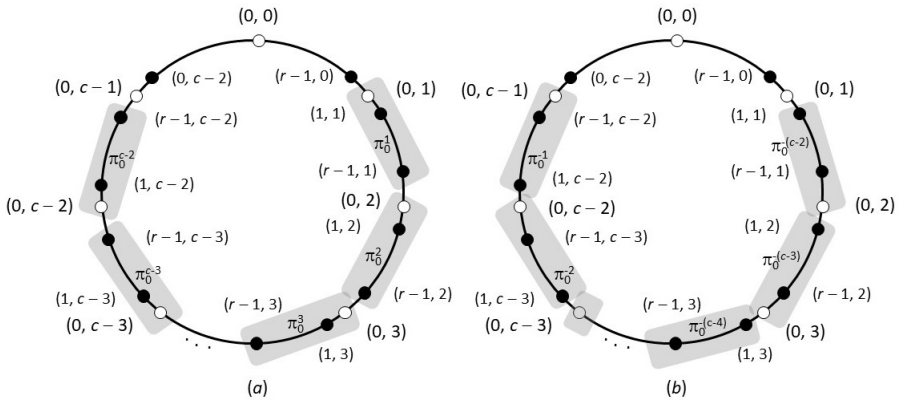

Figure 8: The target vertices of D-walks in Δ_B.

may or may not be less than that using the Type A walk found in Section 6.2 which, by Lemma 6, is at most 2μ or $2\mu + 1$, respectively; that is, these Type B walks need to be checked.

So, henceforth we assume that $c \geq 3$. Consequently, we have a set of $2c - 4$ potential D-walks $\Delta_B = \{\pi_0^i, \pi_0^{-i} : i = 1, 2, \ldots, c-2\}$ for π_0, where for $1 \leq i \leq c-2$:

- π_0^i is the 'clockwise' path of column-edges from $(0,0)$ to (x_0, i) of length $ir + x_0$
- if $x_0 > 0$ then π_0^{-i} is the 'anti-clockwise' path of column-edges from $(0,0)$ to $(x_0, c-1-i)$ of length $(i+1)r - x_0$
- if $x_0 = 0$ then π_0^{-i} is the 'anti-clockwise' path of column-edges from $(0,0)$ to $(0, c-i)$ of length ir.

The target vertices of the paths in Δ_B can be pictured as in Fig. 8 where $\bar{G}_0$ is depicted, with vertices of $\{(0,i) : 0 \leq i \leq c-1\}$ in white. In Fig. 8(a), the target vertices of the paths $\pi_0^1, \pi_0^2, \ldots, \pi_0^{c-2}$ are shown in grey, as are the target vertices of the paths $\pi_0^{-1}, \pi_0^{-2}, \ldots, \pi_0^{-(c-2)}$ in Fig. 8(b). Note that we refer to the walks in Δ_B as 'potential' walks as it may be the case that there is no path in R from some vertex $i \in \{1, 2, \ldots, c-1\}$ of R to y_0. Note how we do not bother about a 'clockwise' D-walk of column-edges to $(0, c-1)$ as the 'anti-clockwise' walk is shorter, and *vice versa* with regard to the vertex $(0,1)$.

For each D-walk in Δ_B, there might be an associated shortest path of row-edges from the target vertex of the D-walk to (x_0, y_0). For $1 \leq i \leq c-2$, we denote the path of row-edges π_1^i (resp. π_1^{-i}) corresponding to π_0^i (resp. π_0^{-i}) to be a path of row-edges isomorphic to a shortest path in R from vertex i (resp. vertex $c-1-i$, if $x_0 > 0$; $c-i$, if $x_0 = 0$) to vertex y_0, if such a path exists; consequently, if the path π^i (resp. π^{-i}) exists then it has length $ir + x_0 + d_R(i, y_0)$ (resp. $(i+1)r - x_0 + d_R(c-1-i, y_0)$, if $x_0 > 0$; $ir + d_R(c-i, y_0)$, if $x_0 = 0$).

So, we have a set $\Delta_B = \{\pi_0^i, \pi_0^{-i} : i = 1, 2, \ldots, c-2\}$ of D-walks for π_0 that we must consider when searching for the shortest walk of Type B from $(0,0)$ to (x_0, y_0). We can actually dispense with some of the walks in Δ_B. Define κ to be the minimal value from I_R (note that $\kappa \leq \frac{c}{2}$).

Lemma 13. *In order to find a shortest Type B D-walk from $(0,0)$ to (x_0, y_0) (bearing in mind the Type A walks already found), we need only consider the D-walks $\{\pi_0^i, \pi_0^{-i} : i = 1, 2, \ldots, \kappa\}$ for π_0.*

Proof. Let $i \in \{1, 2, \ldots, c-2\}$ and consider the walk $\pi^i = \pi_0^i \pi_1^i$, where $\pi_0^i \in \Delta_B$ and π_1^i is as above (we assume π_1^i exists). Suppose that $k \in I_R$ with $i > k \geq 1$. We can obtain a shorter walk from $(0,0)$ to (x_0, y_0) than π^i as follows: take the D-walk $\pi_0^{i-k} \in \Delta_B$ from $(0,0)$ to $(x_0, i-k)$, then the row-edge $((x_0, i-k), (x_0, i))$ and then the path π_1^i. The length of the walk π^i is $ir + x_0 + d_R(i, y_0)$ while the length of the new walk is $(i-k)r + x_0 + 1 + d_R(i, y_0)$. Consequently, there is no need for us to consider the walk π^i. Analogous arguments apply to a walk $\pi^{-i} = \pi_0^{-i} \pi_1^{-i}$ where $\pi_0^{-i} \in \Delta_B$, with the walk $\pi_0^{-(i-k)}$ playing the role of π_0^{i-k}, above (there are two cases: $x_0 > 0$; and $x_0 = 0$). □

We depict a shortened D-walk from $(0,0)$ to $(x_0, 4)$ in $\bar{G}_0$ in Fig. 9 as a dashed line where $\kappa = 3$ and the light grey vertices are the vertices 'ruled out' according to Lemmas 6–9 in Section 6.1, as above, and the white vertices are vertices of the form $(0, i)$. As a result, we can find the length of a shortest Type B walk starting with a D-walk from Δ_B in $O(\kappa c |I_R|)$ time and output a shortest such Type B walk in $O(c(r + \kappa |I_R|))$ time; the length of such a shortest Type B walk is at most $(\kappa + 1)r - 1 + diam(R)$.

Henceforth, we take Δ_B to be the set of walks given by Lemma 13. However, we can prune the set of walks Δ_B even further, depending upon κ and x_0.

Lemma 14. *Suppose that $\kappa \geq 2$ and $x_0 > 0$. In order to find a shortest Type B D-walk from $(0,0)$ to (x_0, y_0) (bearing in mind the Type A walks already found), we need only consider the D-walks*

$$\{\pi_0^1, \pi_0^2, \ldots, \pi_0^{\lceil \frac{\kappa-2}{2} \rceil}, \pi_0^{-1}, \pi_0^{-2}, \ldots, \pi_0^{-\lceil \frac{\kappa-2}{2} \rceil}\}$$

from Δ_B for π_0.

Proof. Let $2 \leq i \leq \kappa - 1$. Consider a D-walk $\pi^{\kappa-i} = \pi_0^{\kappa-i} \pi_1^{\kappa-i}$, where $\pi_0^{\kappa-i} \in \Delta_B$ and $\pi_1^{\kappa-i}$ is a shortest path of row-edges from $(x_0, \kappa - i)$ to (x_0, y_0), in comparison with the D-walk defined as $\pi_0^{-(i-1)} \in \Delta_B$ extended with the row-edge $((x_0, c-i), (x_0, \kappa - i))$ and the path $\pi_1^{\kappa-i}$. The length of the former D-walk is $(\kappa - i)r + x_0 + |\pi_1^{\kappa-i}|$ and the length of the latter is $ir - x_0 + 1 + |\pi_1^{\kappa-i}|$. Hence, if $\kappa \geq 2i$ then we can dispense with the D-walk $\pi_0^{\kappa-i}$ from Δ_B. An analogous argument holds for a walk $\pi^{-(\kappa-i)} = \pi_0^{-(\kappa-i)} \pi_1^{-(\kappa-i)}$, with π_0^{i-1} playing the role of $\pi_0^{-(i-1)}$, above. Hence, we may assume that

$$\Delta_B = \{\pi_0^1, \pi_0^2, \ldots, \pi_0^{\lceil \frac{\kappa-2}{2} \rceil}, \pi_0^{\kappa-1}, \pi_0^{\kappa}, \pi_0^{-1}, \pi_0^{-2}, \ldots, \pi_0^{-\lceil \frac{\kappa-2}{2} \rceil}, \pi_0^{-(\kappa-1)}, \pi_0^{-\kappa}\}.$$

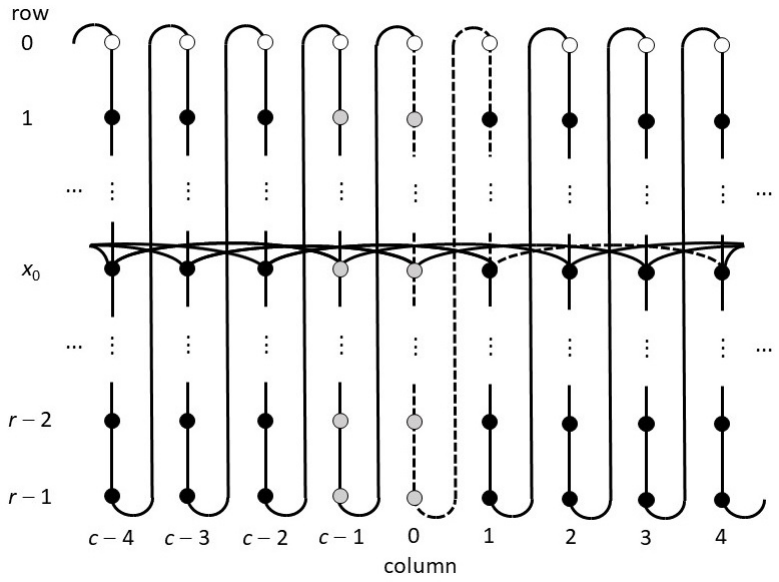

Figure 9: A D-walk to $(x_0, 4)$ in $\bar{G}_0$.

Consider the D-walk $\pi^{\kappa-1} = \pi_0^{\kappa-1}\pi_1^{\kappa-1}$, where $\pi_0^{\kappa-1} \in \Delta_B$ and $\pi_1^{\kappa-1}$ is a shortest path of row-edges from $(x_0, \kappa-1)$ to (x_0, y_0), in comparison with the D-walk defined as the shortest Type A D-walk from $(0,0)$ to $(x_0, c-1)$ (built as in Lemma 7 and Lemma 9) extended with the additional edge $((x_0, c-1), (x_0, \kappa-1))$ and then the path $\pi_1^{\kappa-1}$. The length of the former D-walk is $(\kappa-1)r + x_0 + |\pi_1^{\kappa-1}|$ whereas the length of the latter is at most $\max\{r - x_0, 2\mu - r + x_0\} + 1 + |\pi_1^{\kappa-1}|$. Hence, as $\kappa \geq 2$ and $x_0 > 0$, we can dispense with the D-walk $\pi_0^{\kappa-1}$ from Δ_B. Consider the D-walk $\pi^{\kappa} = \pi_0^{\kappa}\pi_1^{\kappa}$, where $\pi_0^{\kappa} \in \Delta_B$ and π_1^{κ} is a shortest path of row-edges from (x_0, κ) to (x_0, y_0), in comparison with the D-walk defined as the shortest Type A D-walk from $(0,0)$ to $(x_0, 0)$ (built as in Lemma 6 and Lemma 8) extended with the additional edge $((x_0, 0), (x_0, \kappa))$ and then the path π_1^{κ}. The length of the former D-walk is $\kappa r + x_0 + |\pi_1^{\kappa}|$ whereas the length of the latter is at most $\max\{x_0, 2\mu - x_0\} + 1 + |\pi_1^{\kappa}|$. Hence, as $\kappa \geq 2$ and $x_0 > 0$, we can dispense with the D-walk π_0^{κ} from Δ_B. Analogous arguments allow us to dispense with the walks $\pi_0^{-(\kappa-1)}$ and $\pi_0^{-\kappa}$ from Δ_B. In conclusion, we may suppose that

$$\Delta_B = \{\pi_0^1, \pi_0^2, \ldots, \pi_0^{\lceil \frac{\kappa-2}{2} \rceil}, \pi_0^{-1}, \pi_0^{-2}, \ldots, \pi_0^{-\lceil \frac{\kappa-2}{2} \rceil}\}$$

(note that if $\kappa = 2$ then $\Delta_B = \emptyset$). □

We can actually dispense with the original set of walks Δ_B when $\kappa = 1$ and $x_0 > 0$.

Lemma 15. *Suppose that $\kappa = 1$ and $x_0 > 0$. There are no Type B D-walks from $(0,0)$ to (x_0, y_0) needing to be considered (bearing in mind the Type A walks already found).*

PROOF. Consider the D-walk $\pi^1 = \pi_0^1\pi_1^1$, where $\pi_0^1 \in \Delta_B$ and π_1^1 is a shortest path of row-edges from $(x_0, 1)$ to (x_0, y_0), in comparison with the D-walk defined as the shortest Type A D-walk from $(0,0)$ to $(x_0, c-1)$ (built as in Lemma 7 and Lemma 9) extended with the additional edges $((x_0, c-1), (x_0, 0))$ and $((x_0, 0), (x_0, 1))$, and then the path π_1^1. The length of the former D-walk is $r + x_0 + |\pi_1^1|$ whereas the length of the latter is at most $\max\{r - x_0, 2\mu - r + x_0\} + 2 + |\pi_1^1|$. Hence, as $x_0 > 0$, we can dispense with the D-walk π_1^1 from Δ_B. Consider the D-walk $\pi^{-1} = \pi_0^{-1}\pi_1^{-1}$, where $\pi_0^{-1} \in \Delta_B$ and π_1^{-1} is a shortest path of row-edges from $(x_0, c-2)$ to (x_0, y_0), in comparison with the D-walk defined as the shortest Type A D-walk from $(0,0)$ to $(x_0, 0)$ (built as in Lemma 6 and Lemma 8) extended with the additional edges $((x_0, 0), (x_0, c-1))$ and $((x_0, c-1), (x_0, c-2))$, and then the path π_1^{-1}. The length of the former D-walk is $2r - x_0 + |\pi_1^{-1}|$ whereas the length of the latter is at most $\max\{x_0, 2\mu - x_0\} + 2 + |\pi_1^{-1}|$. Hence, as $x_0 > 0$, we can dispense with the D-walk π_1^{-1} from Δ_B. Consequently, when $\kappa = 1$ and $x_0 > 0$ we have that $\Delta_B = \emptyset$. □

Finally, we can dispense with some of the original set of walks Δ_B when $\kappa \geq 2$ and $x_0 = 0$.

Lemma 16. *Suppose that $\kappa \geq 2$ and $x_0 = 0$. In order to find a shortest Type B D-walk from $(0,0)$ to (x_0, y_0) (bearing in mind the Type A walks already found), we need only consider the D-walks*

$$\{\pi_0^1, \pi_0^2, \ldots, \pi_0^{\lfloor \frac{\kappa}{2} \rfloor}, \pi_0^{-1}, \pi_0^{-2}, \ldots, \pi_0^{-\lfloor \frac{\kappa}{2} \rfloor}\}$$

from Δ_B for π_0.

PROOF. If $2 \leq i \leq \kappa - 1$ then by proceeding as we have done throughout, we can dispense with $\pi_0^{\kappa - i} \in \Delta_B$ (by building a new walk using the walk $\pi_0^{-i} \in \Delta_B$) and with $\pi_0^{-(\kappa - i)} \in \Delta_B$ (by building a new walk using the walk $\pi_0^i \in \Delta_B$), so long as $\kappa > 2i$. We can also dispense with $\pi_0^\kappa \in \Delta_B$ (by building a new walk using a Type A walk) and with $\pi_0^{-\kappa} \in \Delta_B$ (by building a new walk using a Type A walk). Consequently, when $\kappa \geq 2$ and $x_0 = 0$, we have that

$$\Delta_B = \{\pi_0^1, \pi_0^2, \ldots, \pi_0^{\lfloor \frac{\kappa}{2} \rfloor}, \pi_0^{-1}, \pi_0^{-2}, \ldots, \pi_0^{-\lfloor \frac{\kappa}{2} \rfloor}\}.$$

□

In summary:

- if $I_R = \emptyset$ then we can find the length of a shortest Type B walk in $\bar{G}_0$ from $(0,0)$ to (x_0, y_0), which is at most $\frac{cr}{2}$, in $O(1)$ time and output an actual shortest walk in time $O(cr)$; and

- if $I_R \neq \emptyset$ then we can find the length of a shortest Type B walk in $\bar{G}_0$ from $(0,0)$ to (x_0, y_0) which is at most $(\lfloor \frac{\kappa}{2} \rfloor + 1)r - 1 + diam(R)$, in $O(\kappa c|I_R|)$ time and output an actual shortest path in time $O(c(r + \kappa|I_R|))$,

where κ is the minimal value from I_R. We reiterate that when we say 'a shortest Type B walk', we mean after taking into account the Type A walks previously considered.

6.4. Our main results

So, our analysis above enables us to prove the following theorems (with all parameters as defined throughout this paper).

Theorem 17. *Suppose that we are given a toroidal semidirect product graph S on $\mathcal{Q} \rtimes \mathcal{G}$, with parameters as per the construction of S and where $I_R \neq \emptyset$, with κ the minimum element of I_R. There is an algorithm that produces a shortest path between any two given vertices that has time complexity $O(r(c + rn) + c\kappa|I_R| + |\mathcal{H}||\Gamma_{\mathcal{H}}|$, and the diameter of S is at most $\max\{2r - 2, (\lfloor \frac{\kappa}{2} \rfloor + 1)r - 1\} + diam(R) + diam(Cay(\mathcal{H}; \Gamma_{\mathcal{H}})) + diam(T)$. In particular, the shortest path algorithm is polynomial in $|\mathcal{G}|$ and polylogarithmic in $|\mathcal{Q}|$.*

Proof. As we stated earlier, because S is vertex-transitive, w.l.o.g. we may assume that our given vertices are $start = (1_e, (1_e, 0, 0))$ and $end = (\mathbf{q}, (\sigma, x_0, y_0))$. From Section 5.2, we need to find a shortest D-walk from $(0,0)$ to (x_0, y_0) in $\bar{G}_0$, expand this to a shortest path in S from $start$ to $end' = (\mathbf{q}, (1_e, x_0, y_0))$ and then extend this path using a shortest path in $Cay(\mathcal{H}; \Gamma_{\mathcal{H}})$ from 1_e to σ.

Our analysis in Section 6.1, our algorithm *ShortestDWalk* in Section 6.2 and a breadth-first search allow us to compute the shortest Type A D-walk from $(0,0)$ to (x_0, y_0) in $\bar{G}_0$ (if it exists) with time complexity $O(r^2 n + c|I_R|)$, and this walk has length at most $2r - 2 + diam(R)$.

Our analysis in Section 6.3 (primarily Lemma 13) allows us to compute the shortest type B D-walk from $(0,0)$ to (x_0, y_0) in $\bar{G}_0$, bearing in mind the Type A D-walks already explored, with time complexity $O(c(r + \kappa|I_R|))$, and the length of such a walk is at most $(\lfloor \frac{\kappa}{2} \rfloor + 1)r - 1 + diam(R)$.

Our chosen D-walk from $(0,0)$ to (x_0, y_0) in $\bar{G}$ can be expanded to a path in S from $start$ to end' with time complexity $O(n)$ (we treat $b_1, b_2, \ldots, b_m$ in the definition of $\mathcal{Q}$ as constants) and then extended to a path from $start$ to end in S with an additional time complexity $O(|\mathcal{H}||\Gamma_{\mathcal{H}}|)$. The length of the resulting path is greater than the length of our chosen D-walk by $diam(Cay(\mathcal{H}; \Gamma_{\mathcal{H}})) + diam(T)$, where $diam(T)$ is the diameter of the torus T from Section 5.1. The results follow. □

An identical analysis but where $I_R = \emptyset$ yields the following result.

Theorem 18. *Suppose that we are given a toroidal semidirect product graph S on $\mathcal{Q} \rtimes \mathcal{G}$, with parameters as per the construction of S and where $I_R = \emptyset$. There is an algorithm that produces a shortest path between any two given vertices that has time complexity $O(r^2 n)$, and the diameter of S is at most $\max\{2r - 2, \frac{cr}{2}\} + diam(R) + diam(Cay(\mathcal{H}; \Gamma_{\mathcal{H}})) + diam(T)$.*

More refined results than Theorem 17 exist in special cases covered by Lemmas 14–16. For example, by Lemma 15, if $\kappa = 1$ (as is the case with cube-connected circulants) and $x_0 > 0$ then we need not worry about looking for Type B walks.

7. Conclusions

We have developed a framework within which we can build a vast range of Cayley graphs of semidirect products of abelian groups, which we call collectively toroidal semidirect product graphs, and we have further designed an efficient shortest-path routing algorithm for any of the graphs defined within our framework; in the process, we show that the diameter of any of our graphs is low with regard to the number of vertices. So, we have demonstrated that our graphs have strong potential for use as interconnection networks (as well as being of interest purely from a combinatorial perspective; of course, the fact that our graphs are Cayley graphs means that they are vertex-transitive too). We present some general comments and directions for further research below.

7.1. More on connectivity

What remains is to fully classify the connectivity of the full class of toroidal semidirect product graphs. In tandem with this, the wide diameter should be investigated, where the *wide diameter* of a graph with connectivity c is the least value w so that given any two vertices of the graph, there are c pairwise vertex-disjoint paths from one vertex to the other so that all paths have length at most w. Of course, the lower the wide diameter of a graph, the intuitively better it is with regard to data broadcasting and fault tolerance.

7.2. Enhancing exchanged hypercubes

We have already mentioned dual-cubes. However, dual-cubes are special cases of the more general exchanged cubes, defined as follows. Given $s, t \geq 1$, the *exchanged cube* $EH(s,t)$ has vertex set $\{0,1\}^{s+t+1}$ so that the vertex $(\mathbf{u}, \mathbf{v}, w) \in \{0,1\}^s \times \{0,1\}^t \times \{0,1\}$ has neighbours:

- $(\mathbf{u}, \mathbf{v}, w) + \mathbf{e}_{s+t+1}$
- $(\mathbf{u}, \mathbf{v}, w) + \mathbf{e}_j$, for $1 \leq j \leq s$, if $w = 0$
- $(\mathbf{u}, \mathbf{v}, w) + \mathbf{e}_{s+j}$, for $1 \leq j \leq t$, if $w = 1$,

where $\mathbf{e}_j$ is the $(s+t+1)$-tuple with 1 in the jth component and 0 elsewhere, and addition is modulo 2. The dual-cube DC_n is isomorphic to $EH(n,n)$. Exchanged cubes originated in [24] and have been subsequently investigated as potential interconnection networks.

A major disadvantage of $EH(s,t)$ is that if $s \neq t$ then $EH(s,t)$ is not even regular never mind vertex-transitive. However, the graph within our framework that is analogous to $EH(s,t)$ is the recursive cubes of rings where r is equal to the least common multiple of s and $s+t$, with $d = s$ and $n = s+t$. Research

should be undertaken to compare such recursive cubes of rings with exchanged hypercubes in the hope that they retain the beneficial properties of exchanged hypercubes but have the added advantage that they are also Cayley graphs.

7.3. Extensions to the framework

Looking at some of the classes of graphs formed using semidirect products but which fail to fit within our framework, as described in Section 3.5, we now suggest directions in which we might extend our framework but so as to retain some sort of control that will enable us to, for example, prove distance properties or develop routing algorithms. Key to our analysis throughout has been the fact that routing in a torus is 'dimensional' and our generators from Γ_S adhere to these dimensions (see the discussion at the end of Section 5.2). Might we be able to work with 'non-dimensional' generators, so that we might capture classes of graphs such as wrapped butterflies? Doing so would complicate the relationship between routing via the $\mathcal{Q}$-edges and dimension-covering in G. Also, might we be able to vary the action of $\mathcal{G}$ on $\mathcal{Q}$ from the current 'wreath-like' action so as to capture classes of graphs such as the supertoroids? Intuitively, to do so would take us out of our 'dimensional' environment. Finally, there is a body of work, of which [6] is representative, on Cayley graphs of abelian groups as potential interconnection networks where these graphs are known as *lattice graphs*. Can we incorporate lattice graphs into our framework (as extensions of the circulant graphs that currently feature)?

References

[1] D. Aguirre-Guerrero, G. Ducoffe, L. Fàbrega, P. Vilà, and D. Coudert. Low time complexity algorithms for path computation in Cayley graphs. *Discrete Applied Mathematics*, 259:218–225, 2019.

[2] S.B. Akers and B. Krishnamurthy. On group graphs and their fault tolerance. *IEEE Transactions on Computers*, C-36(7):885–887, 1987.

[3] B. Alspach. Cayley graphs with optimal fault tolerance. *IEEE Transactions on Computers*, 41(10):1337–1339, 1992.

[4] L. Babai. Chromatic number and subgraphs of Cayley graphs. In Y. Alavi and D.R. Lick, editors, *Theory and Applications of Graphs*, Lecture Notes in Mathematics Vol. 642, pages 10–22. Springer, 1978.

[5] J.-Y. Cai, G. Havas, B. Mans, A. Nerurkar, J.-P. Seifert, and I. Shparlinski. On routing in circulant graphs. In T. Asano, H. Imai, D.T. Lee, S. Nakano, and T. Tokuyama, editors, *Proc. 5th Ann. Int. Conf. on Computing and Combinatorics, COCOON*, Lecture Notes in Computer Science Vol. 1627, pages 360–369. Springer, 1999.

[6] C. Camarero, C. Martinez, and R. Beivide. Lattice graphs for high-scale interconnection topologies. *IEEE Transactions on Parallel and Distributed Systems*, 26(9):2506–2519, 2015.

[7] C. Camarero, C. Martinez, E. Vallejo, and R. Beivide. Projective networks: topologies for large parallel computer systems. *IEEE Transactions on Parallel and Distributed Systems*, 28(7):2003–2016, 2017.

[8] G.E. Carlsson, J.E. Cruthirds, H.B. Sexton, and C.J. Wright. Interconnection networks based on a generalization of cube-connected cycles. *IEEE Transactions on Computers*, C-34(8):769–772, 1985.

[9] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to Algorithms*. MIT Press, 2009.

[10] D. Coudert and G. Ducoffe. Data center interconnection networks are not hyperbolic. *Theoretical Computer Science*, 639:72–90, 2016.

[11] W.J. Dally and B. Towles. *Principles and Practices of Interconnection Networks*. Elsevier, 2004.

[12] J.D. Dixon and B. Mortimer. *Permutation Groups*. Springer, 1991.

[13] R.N. Draper. Supertoroidal networks. Technical Report SRC-TR-90-005, Super-Computing Research Center, IDA, MD, 1990.

[14] R.N. Draper and V. Faber. The diameter and mean diameter of supertoroidal networks. Technical Report SRC-TR-90-004, Super-Computing Research Center, IDA, MD, 1990.

[15] D.B.A. Epstein, M.S. Paterson, J.W. Cannon, D.F. Holt, S.V. Levy, and W.P. Thurston. *Word Processing in Groups*. A.K. Peters, 1992.

[16] J.C. George, A. Khodkar, and W.D. Wallis. *Pancyclic and Bipancyclic Graphs*. Springer, 2016.

[17] R. Hammack, W. Imrich, and S. Klavžar. *Handbook of Product Graphs*. CRC Press, 2011.

[18] M.C. Heydemann. Cayley graphs and interconnection networks. In G. Hahn and G. Sabidussi, editors, *Graph Symmetry*, pages 167–224. Kluwer, 1997.

[19] L.-H. Hsu and C.-K. Lin. *Graph Theory and Interconnection Networks*. Taylor and Francis, 2009.

[20] S. Lakshmivarahan, J.-S. Jwo, and S.K. Dhall. Symmetry in interconnection networks based on Cayley graphs of permutation groups: a survey. *Parallel Computing*, 19(4):361–407, 1993.

[21] F.T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays. Trees. Hypercubes*. Morgan Kaufmann, 1992.

[22] X. Li and Y. Mao. *Generalized Connectivity of Graphs*. Springer, 2016.

[23] Y. Li and S. Peng. Fault-tolerant routing and disjoint paths in dual-cube: a new interconnection network. In *Proc. of 8th Int. Conf. on Parallel and Distributed Systems, ICPADS*, pages 315–322. IEEE, 2001.

[24] P.K.K. Loh, W.J. Hsu, and Y. Pan. The exchanged hypercube. *IEEE Transactions on Parallel and Distributed Systems*, 16(9):866–874, 2005.

[25] W. Mader. Über den Zusammenhang symmetrischer Graphen. *Archiv der Mathematik*, 21(1):331–336, 1970.

[26] W. Mader. Eine Eigenschaft der Atome endlicher Graphen. *Archiv der Mathematik*, 22(1):333–336, 1971.

[27] W. Mader. Minimale n-fach kantenzusammenhängende Graphen. *Mathematische Annalen*, 191(1):21–28, 1971.

[28] H. Mokhtar. Cube-connected circulants as efficient models for interconnection networks. *Journal of Interconnection Networks*, 17(3-4):article 1741007, 2017.

[29] H. Mokhtar and S. Zhou. Recursive cubes of rings as models for interconnection networks. *Discrete Applied Mathematics*, 217(3):639–662, 2017.

[30] F. Preparata and J. Vuillemin. The cube-connected cycles: a versatile network for parallel computation. *Communications of the ACM*, 24(5):300–309, 1981.

[31] I.A. Stewart. Multiswapped networks and their topological and algorithmic properties. *Journal of Computer and System Sciences*, 79(8):1269–1286, 2013.

[32] I.A. Stewart. Sufficient conditions for Hamiltonicity in multiswapped networks. *Journal of Parallel and Distributed Computing*, 101:17–26, 2017.

[33] I. Stojmenovic. Multiplicative circulant networks topological properties and communication algorithms. *Discrete Applied Mathematics*, 77(3):281–305, 1997.

[34] Y. Sun, P. Cheung, and X. Lin. Recursive cube of rings: a new topology for interconnection networks. *IEEE Transactions on Parallel and Distributed Systems*, 11(3):275–286, 2000.

[35] M.E. Watkins. Some classes of hypoconnected vertex-transitive graphs. In W.T. Tutte and C.St.J.A. Nash-Williams, editors, *Recent Progress in Combinatorics*, pages 323–328. Academic Press, 1969.

[36] M.E. Watkins. Connectivity of transitive graphs. *Journal of Combinatorial Theory*, 8:23–29, 1970.

[37] F.L. Wu, S. Lakshmivarahan, and S.K. Dhall. Routing in a class of Cayley graphs of semidirect products of finite groups. *Journal of Parallel and Distributed Computing*, 60(5):539–565, 2000.

[38] W. Xiao and B. Parhami. A group construction method with applications to deriving pruned interconnection networks. *IEEE Transactions on Parallel and Distributed Systems*, 18(5):637–643, 2007.

[39] W. Xiao, B. Parhami, W. Chen, M. He, and W. Wei. Fully symmetric swapped networks based on bipartite cluster connectivity. *Information Processing Letters*, 110(6):211–215, 2010.

[40] J. Xu. *Topological Structure and Analysis of Interconnection Networks*. Kluwer, 2001.

[41] S. Zhao and R. Hao. The generalized three-connectivity of two kinds of Cayley graphs. *Computer Journal*, 62(1):144–149, 2018.

[42] S. Zhou, L. Chen, and J.-M. Xu. Conditional fault diagnosability of dual-cubes. *International Journal of Foundations of Computer Science*, 23(8):1729–1747, 2012.